Universidade do Minho
Escola de Engenharia

Miguel Ferreira Rego

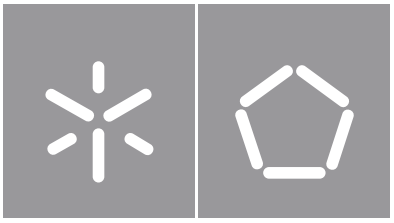
Coordenador de uma rede *Low Power Wide Area*

janeiro de 2019

Coordenador de uma rede *Low Power Wide Area*

Miguel Rego

UMinho | 2019



Universidade do Minho
Escola de Engenharia

Miguel Ferreira Rego

Coordenador de uma rede *Low Power Wide Area*

Dissertação de Mestrado
Engenharia Eletrónica Industrial e Computadores

Trabalho efetuado sob a orientação do(a)
Professor Doutor Jorge Cabral

janeiro de 2019

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Agradecimentos

Agradeço aos meus pais, por estarem desde sempre ao meu lado, pelo sacrifício de nunca me faltar nada e ter sempre acesso a tudo tanto no meu percurso académico como na vida. Sem eles nunca seria possível desenvolver esta dissertação e realizar os objetivos até aqui alcançados. Um enorme obrigado a eles, que nestas poucas palavras chegam para agradecer o quanto estou grato por todos os sacrifícios.

Agradeço ao "meu grupo" que me acompanham desde o início do meu percurso universitário, por todos os momentos e por uma amizade cada vez maior. Obrigado a eles pela motivação e incentivo disponibilizados ao longo deste percurso. Um agradecimento especial a todos os membros do laboratório ESRG instalados no IBS, pelo enorme apoio no desenvolvimento da minha dissertação. Obrigado a eles por todo o conhecimento transmitido, toda a ajuda e motivação imprescindíveis. Agradeço ao Sérgio e à Filipa, que me acompanham diariamente, pela possibilidade de discutir e criar novas ideias juntos, pela camaradagem e pelos momentos que o futuro "fragmentado" reserva. Um obrigado enorme à Catarina pelo apoio, paciência e motivação disponibilizada, pela constante presença, de nunca desistir de mim, pelos sacrifícios, compreensão e por toda ajuda fornecida.

Agradeço ao professor Cabral pela ajuda disponibilizada, pelas oportunidades, pela motivação e por fornecer tudo para que fosse possível realizar esta dissertação. Deixo também um obrigado ao curso de Engenharia Eletrónica Industrial e de Computadores, aos docentes e alunos com o qual tive o prazer de encontrar ao longo destes anos, são eles também parte desta dissertação.

Um muito obrigado a todos eles e poucas palavras nunca irão chegar para exprimir a minha gratidão.

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

Resumo

Nos últimos dez anos, o mundo atual tem presenciado uma evolução exponencial ao nível da tecnologia. A expansão da Internet possibilitou o surgimento de conexões entre diversas partes do globo e a capacidade de integrar esta funcionalidade em vários dispositivos com o qual lidamos no quotidiano. Estes, constituem parte da indústria *Internet of Things* (IoT), onde um número grande de dispositivos estão ligados entre si com o intuito de recolher dados e transmitir estes entre si e a Internet.

Muitas das aplicações desenvolvidas atualmente por partes de grandes empresas têm como foco a monitorização remota, através de um serviço *cloud*, de grandes áreas geográficas (cidades, campos agrícolas ou florestas) recorrendo a dispositivos de baixo consumo energético, custo de produção reduzido e com a mínima manutenção necessária. As redes *Low Power Wide Area* (LPWAN), surgem assim em resposta a este âmbito de aplicações cumprindo os requisitos impostas pelas aplicações do século atual: dispositivos com grande alcance e elevada longevidade.

Os incêndios florestais são um dos grandes problemas da atualidade em diversos pontos do globo sendo que Portugal registou um elevado número de vitimas e infraestruturas no ano 2017 afetadas por parte deste problema global. Contudo, catástrofes deste género não possuem meios adequados na prevenção dos mesmos de modo a reduzir o impacto destas.

A presente dissertação foca-se na criação de um gateway, integrado numa rede LoRaWAN (tecnologia LPWAN), com a capacidade de receber informação proveniente de dispositivos equipados com sensores direcionados à prevenção de incêndios. Estes dados são posteriormente reencaminhados para um servidor de rede onde são armazenados e analisados por parte de uma aplicação *web* desenvolvida tendo como alvo as unidades de combate a incêndios. Desta modo, é possivel posicionar os dispositivos da rede, nós e gateway, recorrendo à API Google Maps e consequentemente monitorizar a rede remotamente, possibilitando assim um alerta antecipado de fogos florestais.

Palavras-Chave: LPWAN, LoRa, LoRaWAN, Incêndios Florestais.

Abstract

In the last ten years, the world has witnessed an exponential evolution in technology. The expansion of the Internet is one of the biggest evolutions which brought the connection between different locations around the world. Nowadays this feature can be embedded in small devices that are able to send information over the Internet. These devices are part of a big industry called Internet of Things (IoT), where a large number of them are connected to each other and are capable of gathering information (with sensors) and later send it to another device or to the Internet.

Many of recent applications developed by companies worldwide aim to build networks capable of monitoring remotely large geographical areas (cities, farms or forests) while beign low energy consumption and having low cost production. The Low Power Wide Area Network technologies are the solution for these kinds of applications, by fulfilling the requirements imposed by the modern world companies: devices with long range and low power consumption.

Wildfires are one of the biggest problems of today's world. However some countries are more affected than others by this problem, which is the case of Portugal over the last years. In 2017, Portugal recorded the highest number of victims and infrastructures affected by this worldwide problem: the wildfires. Many countries already developed many technologies in order to solve this problem, however none of them proved to be the best solution due to being expensive and not being suffienctly autonomous.

This dissertation aims to create a gateway able to make part of a LoRaWAN network (LPWAN technology), capable of receiving information from devices equipped with sensors for proper wildfire detection. The information is later forwarded to the network server in order to be stored and analysed by a developed web application. The application proposes to help the firemen by featuring a map (Google Maps API) containing the position of all the network devices (gateway and nodes) and can also display all the information received in real-time. With these three elements (devices, network server and application) the firefighters can monitor the forests remotely, preventing the fire from spreading and alerting all units in advance.

Keywords: LPWAN, LoRa, LoRaWAN, Wildfires.

Índice

Lista de Figuras

Lista de Tabelas

Lista de Listagens

Lista de Equações

Lista de Acrónimos

ABP	*Activation By Personalisation*.
API	*Application Programming Interface*.
BW	*Bandwidth*.
CF	*Carrier Frequency*.
CR	*Coding Rate*.
CSS	*Chirp Spread Spectrum*.
FSK	*Frequency Shift Key*.
HAL	*Hardware Abstraction Layer*.
IoT	*Internet Of Things*.
LoRa	*Long Range*.
LPWA	*Low Power Wide Area*.
LPWAN	*Low Power Wide Area Network*.
OS	Sistema Operativo.
OTAA	*Over The Air Activation*.
PCB	*Printed Circuit Board*.
RSSI	*Received Signal Strength Indicator*.
SF	*Spreading Factor*.
SNR	*Signal to Noise Ratio*.
SPI	*Serial Peripheral Interface*.
TP	*Transmission Power*.

TTN *The Things Network*.

UART *Universal Asynchronous Receiver/Transmitter*.

Capítulo 1

Introdução

O termo IoT (*Internet of Things*) surge como fruto de uma apresentação para a *Protect&Gamble* através Kevin Ashton em 1999. Porém, o conceito em si remota para o início de 1926 a partir de uma entrevista realizada ao conceituado cientista Nikola Tesla, para a revista Colliers. Tesla idealizou um mundo convertido num cérebro humano em que todos os dispositivos *wireless* estariam conectados entre sim, possibilitando uma comunicação em tempo real independentemente das distâncias. A ideia apresentada, tem vindo a ser estudada ao longo dos últimos anos levando ao aparecimento de novas tecnologias com a capacidade de modificar o quotidiano e a maneira como é realizada a interação com o mundo físico.

A evolução tecnológica crescente, levou ao surgimento de dispositivos IoT dotados de sensores adequados ao meio e como principal característica: a capacidade de se conectarem à Internet. Em 1980 foi produzida, na Universidade de Carnegie Mellon, um dos primeiros produtos que utilizava a tecnologia mencionada. Uma simples máquina de refrigerantes que, uma vez conectada à Internet, oferecia a funcionalidade de permitir ao utilizador verificar se uma determinada bebida estava disponível e consequentemente fresca. O sucesso das IoTs na última década foi crescendo exponencialmente, bem como o seu uso por grandes empresas, como é o caso da LG na criação de um frigorífico ligado à Internet e da *Ambient Orb*, a partir de um *spin-off* da MIT, capaz de informar o seu utilizador acerca de previsões meteorológicas, tráfego rodoviário bem como dados relativos ao mercado atual [1].

A exigência do mercado atual pela necessidade de dispositivos de baixo consumo energético, com a capacidade de conectarem entre si (M2M), possibilitou o surgimento de tecnologias LP-WAN (*Low Power Wide Area Networks*) inseridas em aplicações IoT. Estas redes, oferecem a capacidade de comunicação a longas distâncias entre os dispositivos, dotados de sensores, da rede que posteriormente enviam a informação, após um pré-processamento mínimo (restrições de desempenho e energia) dos dados obtidos acerca do meio, para um servidor web onde é armazenada e futuramente processada. A utilização de protocolos de comunicação adequados permite, também, a obtenção das métricas já mencionadas, sendo desenvolvidos com o intuito de reduzir consumos, fornecer segurança e eficiência nas comunicações.

1.1 Motivação

As IoTs, são atualmente reconhecidas como sendo uma das tecnologias responsáveis pelo surgimento da quarta revolução industrial, a indústria 4.0. A utilização de dispositivos conectados à Internet e entre si, *Machine to Machine* (M2M), tem vindo a tornar-se cada vez mais comum, oferecendo um vasto leque de aplicações como relógios (*smartwatches*), com a capacidade de enviar e receber informações. Os números estimados pela Cisco, revelam um futuro dominado pelas tecnologias IoT, prevendo que em 2020 cerca de 50 mil milhões de dispositivos estarão conectados à Internet, pressupondo que o número da população mundial nesse ano será de 7,6 mil milhões [2].

Atualmente, grande parte dos dispositivos IoT realiza a sua comunicação através de Wi-Fi, redes telefónicas ou ZigBee. Porém, estas tecnologias apresentam um custo elevado com o intuito de alcançar maior alcances e baixos consumos energéticos em certas aplicações IoT. Neste âmbito, o uso de tecnologias LPWAN apresentam-se como uma solução para este problema. Futuramente, as mais diversas cidades do globo possuirão dispositivos inseridos em redes de longo alcance capazes de atuar consoante a informação recolhida (*smart cities*), áreas extensas de campos agrícolas serão monitorizados, através de sensores, informando o proprietário do estado dos seus produtos. São inúmeras as soluções oferecidas pelas tecnologias LPWAN no auxílio aos mais diversos problemas do nosso quotidiano, o que conduz a um investimento nesta tecnologia sendo estimado que em 2025 11% das conexões IoT serão realizadas a partir de uma rede LPWA [3].

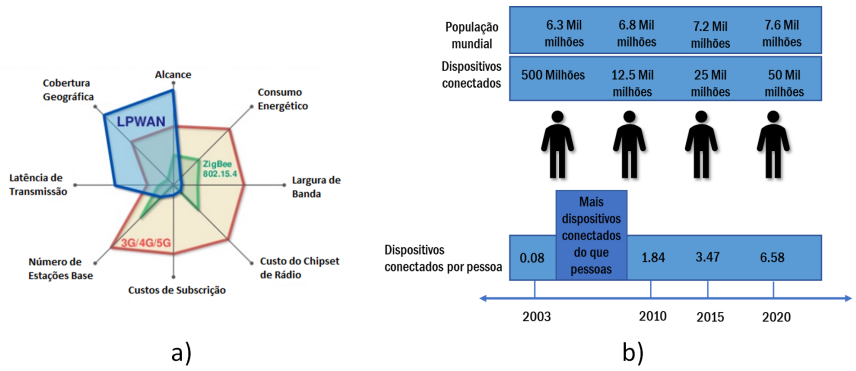


Figura 1.1: A Figura (a) representa uma comparação das diversas tecnologias existentes no mercado, realçando os seus pontos fortes. Relativamente à Figura (b), demonstra as estimativas do número de dispositivos que irão estar conectados, futuramente, à Internet, comparando com a população mundial. (Adaptado de [4])

Um dos problemas mais mediáticos da atualidade em Portugal são os incêndios florestais que todos os anos representam uma ameaça e ao qual não existe uma prevenção eficaz. Os danos causados refletem-se principalmente na área ardida, tendo, contudo, nos últimos anos o número de infraestruturas afetadas aumentado tal como o número de vítimas, provocadas

pelos incêndios. O Instituto da Conservação da Natureza e das Florestas (ICNF), responsável por realizar o balanço dos incêndios florestais em Portugal, realça o ano de 2003 que possui um dos piores registos de área ardida, 425.839 hectares. A ICNF apresenta também dados relativamente ao prejuízo, monetário, causado por estas catástrofes, onde revela gastos associados ao ano de 2005 na ordem dos 750 milhões de euros. O ano de 2017 apresenta-se, baseado em estimativas provisórias, como o pior ano de incêndios florestais atingindo recordes em área ardida, 442.418 hectares, bem como o prejuízo socioeconómicos, que se encontra em fase de estudo, mas que demonstra ate ao momento gastos que ja ultrapassam o ano de 2005 [5].

Atualmente Portugal não dispõe de meios de prevenção equipados com tecnologia contemporânea. Em algumas partes do mundo, encontram-se implementadas diversas técnicas relativamente à deteção de incêndios: *EYEfi SPARC* sensores óticos com intuito de detetar fogos florestais, encontra-se em funcionamento na Austrália; *ForestWatch* que se encontra implementado em diversas partes do globo (principalmente em África), recorre a câmaras térmicas com a capacidade de detetar fumos e a luminosidade produzida pelo fogo a longo alcance (16-20km). Porém, estas soluções são dispendiosas e não se revelam totalmente eficientes. Uma solução para este problema, incêndios florestais, com um potencial elevado é utilização de tecnologia LPWAN capaz de abranger grandes áreas florestais e garantir uma "previsão" atempada de fogos florestais [6].

1.2 Enquadramento

Deste modo, o projeto na qual esta dissertação se encontra inserida, aborda o problema descrito através da criação de uma rede baseada em tecnologia LPWAN com a capacidade de alertar antecipadamente as unidades de combate a incêndios. Através da utilização de nós sensores, equipados com dispositivos adequados à deteção de temperatura, velocidade do vento e humidade, é possível retirar informação do meio e armazenar esta. Posteriormente, a informação retida é enviada para o coordenador de rede com o objetivo de a retransmitir para os microsserviços web, onde é analisada e consequentemente retiradas as respetivas conclusões. Esta rede, como já mencionada, recorre a uma das tecnologias LPWAN presentes no mercado, sendo neste caso a LoRaWAN.

O projeto encontra-se divido em três partes distintas: nós sensores, coordenador de rede e micro-serviços web. Como ja foi mencionado apenas será focado o gateway da rede, sendo este o tema da dissertação em questão. Os nós da rede e de acordo com o âmbito ja mencionado, são responsáveis por obter informações do meio em que estão inseridos, com o auxilio de sensores adequados na deteção de fogos, tal como necessitam de respeitar as métricas introduzidas pelas tecnologias LPWAN: baixo consumo energético através da redução no tempo de funcionamento do nó e em simultâneo um longo alcance adequado à longevidade do dispositivo. O gateway da

rede, tem como função receber a informação proveniente dos nós para posteriormente, esta, ser transmitida para o servidor de onde é analisada. Existe também a possibilidade de comunicação inversa, ou seja, o servidor de rede transmitir informação para um nó com o intuito de este agir como atuador em aplicações onde seja necessario realizar ações por parte dos *nós*. Por último, o servidor de rede tem como função receber a informação transmitida pelo gateway, sendo também capaz de enviar dados para o mesmo, e incorporar aplicações destinadas a analisar os dados e atuar consoante o resultado obtido pelos mesmos, tendo como exemplo o âmbito dos incêndios esta informação é essencial para alertar unidades de combate a incêndio antecipadamente.

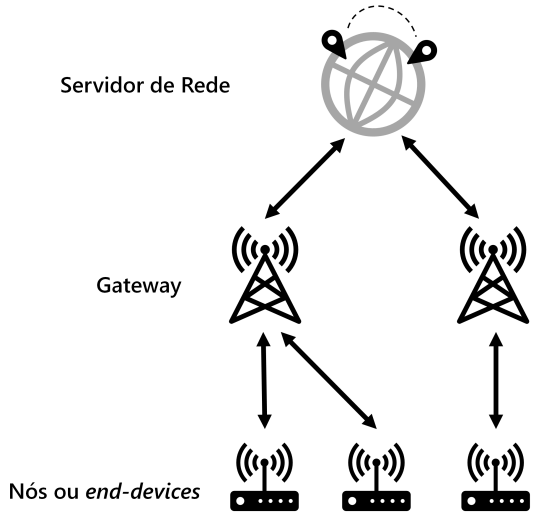


Figura 1.2: A imagem representa o exemplo de uma rede LPWAN.

1.3 Objetivos

Neste capítulo são abordados os principais objetivos da dissertação. Assim, os seguintes pontos representam as metas a atingir relativamente ao coordenador:

- Estudo das tecnologias LPWAN, com o intuito de obter uma análise breve, realçando as características principais, sendo posteriormente realizada uma comparação entre as tecnologias abordas de modo a justificar a decisão por uma rede LoRaWAN;
- Análise de diversos gateways mais utilizados no mercado atual, tendo como objetivo estudar a construção, a nível de hardware, de modo a compreender quais as características que os distinguem e como estas podem fazer parte do gateway final;
- Análise da tecnologia LoRaWAN através da abordagem ao nível da segurança, métodos de inserção de nós, modulação utilizada, arquitetura da rede, componentes e casos de uso;
- Construção de um gateway LoRaWAN de raiz com a capacidade de realizar as funcionalidades básicas;

- Equipar o gateway com diferentes métodos de comunicação de dados, para com o servidor de rede, para situações onde existe a falha de um destes - redundância nas comunicações de dados;
- Estudo e integração do *firmware* desenvolvido pela Semtech que possui a capacidade de receber mensagens provenientes dos nós procedendo posteriormente ao envio destas para o servidor de rede;
- Alteração do *firmware* fornecido pela Semtech, tendo como objetivo incorporar novas funcionalidades de modo a lidar com possíveis falhas nas comunicações com o servidor de rede;
- Realização de testes relacionados com o alcance da rede (indoor e outdoor), possíveis erros na receção de pacotes;
- Estudo dos resultados obtidos e de que forma estes podem ser melhorados;

1.4 Organização da Dissertação

A dissertação em questão encontra-se dividida em seis capítulos distintos, sendo a estrutura definida com o objetivo de fornecer ao leitor uma melhor compreensão do documento em questão.

Assim, o capítulo 1 pretende introduzir e enquadrar o tema desta dissertação, onde são apresentadas as motivações, objetivos e contribuições do projeto desenvolvido.

O capítulo 2, tem como intuito abordar os conceitos básicos necessários ao longo do documento, tendo como foco principal as rede LPWAN. Ainda no capítulo 2, é elaborada uma revisão do estado da arte relativo as tecnologias LPWAN e gateways existentes, sendo apresentado uma conclusão para ambos. O capítulo aborda também em grande detalhe a tecnologia LoRaWAN, sendo esta utilizada no desenvolvimento do projeto.

No capítulo 3, é apresentada as especificações do sistema a qual engloba os requisitos e arquitetura do mesmo. Ainda no capítulo é abordada a plataforma *The Things Network* (TTN) através de uma análise dos serviços utilizados presentes nesta e a escolha do encapsulamento presente no gateway.

O capítulo 4, encontra-se subdividido em duas partes distintas: hardware e software. O primeiro subcapítulo é referente ao desenvolvimento do gateway no que toca a hardware, sendo posteriormente analisado o software presente no mesmo. Ainda no subcapítulo de software, é realizada a análise relativa à plataforma web.

O capítulo 5, destina-se a testes e à respetiva análise dos resultados obtidos a partir dos mesmos. Os testes desenvolvidos focam-se no alcance da rede, receção de pacotes e interpretação dos dados através da aplicação web.

Por ultimo, o capítulo 6 aborda as conclusões da presente dissertação desenvolvida e também as perspetivas futuras do trabalho desenvolvido.

Capítulo 2

Conceitos fundamentais e Estado da Arte

Nesta dissertação pretende-se, assim, implementar um gateway enquadrado numa rede de baixo consumo e longo alcance (LPWAN) baseado em tecnologia LoRaWAN. Porém, é necessário ter em consideração diversos fatores no dimensionamento da rede, que influenciam todo o funcionamento da mesma. Deste modo, é realizada uma abordagem extensa e pormenorizada destes com o intuito de uma melhor compreensão, sendo assim, a segunda parte do presente capítulo destinada aos fatores que influenciam o *design* de uma rede LPWA. Previamente e como primeira parte deste capítulo, são abordados conceitos fulcrais, como topologia de rede, com o objetivo de clarificar os restantes capítulos onde surgem referências a estes conceitos.

O número de tecnologias LPWAN presentes no mercado tem vindo a crescer exponencialmente, tornando difícil a tarefa de decidir por qual optar. Assim, a penúltima parte do capítulo realça as diversas tecnologias, através de uma breve análise, identificando as vantagens e desvantagens comparativamente ao LoRaWAN. Numa última abordagem, é realizada uma conclusão relativamente à escolha da tecnologia, justificando o porque de esta ser adequada para o projeto em causa.

Após a escolha da tecnologia, é necessario ter em consideração os gateways existentes no mercado que possuem suporte LoRaWAN. Deste modo, a ultima parte deste capítulo tem como intuito realizar uma abordagem aos diferentes e principais coordenadores desenvolvidos por diversas empresas no mercado das tecnologias LPWAN. Por último, é elaborada um breve resumo dos gateways analisados justificando a decisão de optar pela criação de um *gateway*.

2.1 Low Power Wide Area Network

As *Low Power Wide Area Networks* (LPWANs) surgem como complemento e melhoria das redes telefónicas e sem fios de curto alcance, respeitando os requisitos de aplicações IoT. Apresentam um leque de características únicas, como a possibilidade de dispositivos comunicarem a longas distâncias com um baixo consumo energético (através de medidas implementadas) e *bitrate*

baixo. É previsto um enorme impacto destas tecnologias no mercado, sendo estimado que um quarto dos 30 mil milhões de dispositivos IoT no futuro vão estar conectados à Internet a partir de uma rede LPWA, como ja foi mencionado anteriormente.

Estas tecnologias tem uma particularidade única que são os *tradeoffs* apresentados, diferenciando-se das restantes tecnologias tradicionais já existentes na área das IoT. As comunicações wireless de curto alcance (ZigBee ou Bluetooth) e Wi-Fi, possui algumas restrições a nível de alcance e consumo energético. O alcance destas é limitado a poucas centenas de metros, deixando de fora as exigências do mercado atual e impedindo que estes dispositivos sejam posicionados em "qualquer lugar". As redes telefónicas por outro lado oferecem um alcance alargado, contudo o consumo por parte destas tecnologias coloca em causa o "tempo de vida" das baterias incorporadas nos dispositivos.

Assim as redes LPWA, apresentam-se como principais soluções relativamente às métricas de consumo energético e alcance, oferecendo aos dispositivos mais de dez anos de "vida" e distâncias na ordem dos quilómetros na comunicação entre dispositivos. O baixo *data rate* e a alta latência permite obter as vantagens mencionadas, porém, restringe a utilização das LPWAN em algumas aplicações onde é exigido velocidade na transmissão de dados, sendo o enquadramento destas tecnologias indicado para as situações onde são toleráveis os atrasos no envio de dados entre outros pontos quer serão abordados ao longo do presente documento [7].

2.2 Topologias de Rede

A topologia de uma rede determina o modo como os diversos elementos de uma rede estão interligados entre si. Redes que apresentam um curto alcance geralmente recorrem a uma topologia *mesh*, com o intuito de aumentar a cobertura da rede, concebendo a possibilidade do sinal proveniente do emissor percorrer diversos "caminhos", em caso de falha de algum elemento, dotando a rede de alguma flexibilidade. Porém, este tipo de abordagem provoca atrasos na propagação do sinal, sendo que este problema é geralmente corrigido através do aumento da taxa do envio de dados [8].

Em redes LPWA o cenário é diferente do abordado no parágrafo anterior, neste tipo de redes é usual uma topologia em estrela, onde os nós comunicam com um sistema central ou gateway. Este tipo de topologia facilita a introdução de novos elementos e assegura o funcionamento de toda a rede em situações de falha ou avaria de um dos nós. O gateway é responsável pelo envio da informação recebida pelos nós para um servidor de rede através de tecnologias como Ethernet ou Wi-Fi. A Figura seguinte representa os dois tipos de topologias mencionadas de forma a obter uma melhor compreensão das mesmas [7].

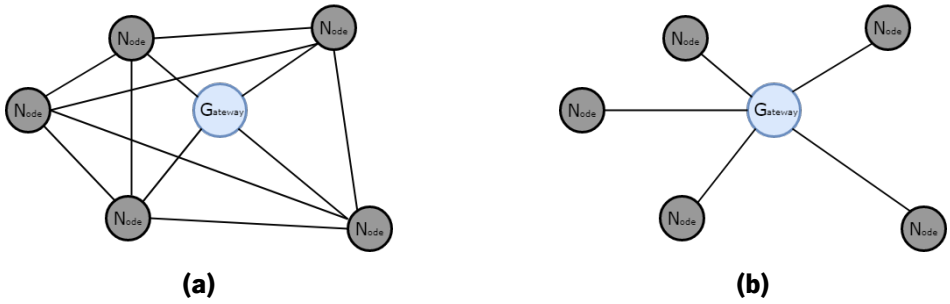


Figura 2.1: Exemplo de duas topologias de rede: (a) Topologia *Mesh*, os diversos nós podem conectar-se entre si e também ao coordenador; (b) Topologia Estrela, cada nó encontra-se apenas ligado ao *gateway*.

2.3 Design de uma rede LPWA

Existem diversos fatores que influenciam na escolha de uma rede LPWA: segurança, qualidade do serviço (QoS), cobertura da rede, consumo energético, custos, etc. Cada aplicação que recorre à utilização de tecnologias LPWAN, necessita de ter em consideração estes parâmetros ao longo do design da rede de modo a obter a solução ideal para a aplicação a que é destinada a rede. Assim o presente capítulo aborda os 6 principais fatores, apresentando uma descrição breve de cada e demonstrando a influencia que possuem na escolha e design de uma rede LPWAN.

2.3.1 Cobertura de Rede

Cobertura e alcance de rede são parâmetros que diferem entre si significativamente. Enquanto o alcance pode ser considerado o máximo que a rede consegue atingir, em termos de distâncias, a cobertura por outro lado apresenta-se como sendo a área abrangida pela rede tendo em conta qualquer obstáculo que impede esta de atingir o seu alcance máximo. Em redes LPWA, é necessário ter em consideração a cobertura em tempo real, ou seja, a probabilidade de uma mensagem ser recebida no momento em que é enviada.

O alcance é uma das principais propriedades das rede LPWAN que torna possível o posicionamento de dispositivos, auxiliados de baixo consumo, a grandes distâncias de um gateway conduzindo também a uma maior mobilidade (possibilidade dos dispositivos mudarem a sua localização num área alargada). Estas redes possibilitam uma cobertura outdoor, indoor e underground seja em áreas rurais ou urbanas, devido à utilização de frequências baixas (Sub-GHz) que garante a ultrapassagem/penetração de diversos obstáculos (árvores ou paredes) possibilitando o posicionamento de nós em locais remotos.

O *link budget* é utilizado no design da rede encontrando-se diretamente ligado com o alcance e a cobertura da mesma. Esta métrica, determina (como apresenta a equação 2.1) quais os

valores necessários a atribuir aos ganhos das antenas, níveis de potência nas transmissões e a sensibilidade do recetor ao longo da comunicação [9].

$$PotenciaReceptor(dBm) = PotenciaTransmissor(dBm) + Ganhos(dB) - Perdas(dB) \tag{2.1}$$

Equation 2.1: Determinação do *link budget*.

2.3.2 Custos Energéticos

Um dos requisitos fundamentais de uma rede LPWA é a poupança energética por parte dos dispositivos presentes na rede. São diversos os fatores que influenciam os gastos da bateria: o tempo de funcionamento (*On-time*) do dispositivo, os consumos nas transmissões, o próprio sensor durante aquisição de informação (nós sensoriais), a quantidade de dados que são enviados e o bit rate associado. A bateria dos dispositivos, no que toca à sua duração, revela-se difícil de determinar podendo durar meses ou anos consoante os fatores mencionados. A topologia de rede em estrela, utilizada na redes LPWAN, revela-se também um fator decisivo no consumo energético dos dispositivos. [10].

2.3.3 Qualidade do Serviço (QoS)

Normalmente a qualidade de serviço é associada à velocidade de receção de mensagens bem como a probabilidade de êxito das mesmas, tal como a fiabilidade da rede. Para garantir esta qualidade são requisitados *trade-offs* associados, um atraso na receção de uma mensagem geralmente tem como origem um congestionamento na rede, por outro lado aumentando a rede (adição de gateways) pode solucionar o problema levando a um acréscimo da qualidade do serviço, mas possuindo um custo associado. É de salientar que nenhum sistema é capaz de assegurar uma qualidade de serviço total, porém é possível aumentar a probabilidade de garantir a qualidade no serviço adequando ao contexto e requisitos da rede.

A fiabilidade da rede representa um fator fulcral na obtenção da qualidade do serviço, sendo determinada pela percentagem de mensagens recebidas com sucesso. As falhas nas transmissões podem ser causadas por defeitos no hardware, congestionamentos na rede ou interferências e portanto devem ser evitados de modo a impedir qualquer perda de informação e consequentemente garantindo uma maior qualidade no serviço. Também associado a possíveis problemas que influenciam este fator, é a utilização de um gateway apenas em locais que o numero destes diapositivos se encontra limitado, encarregando-o de lidar ou comunicar com uma quantidade elevada de nós [11].

2.3.4 Segurança

Os elementos de segurança de uma rede LPWA são diversos: a rede fornece serviços de autenticação aos dispositivos verificando se estes pertencem à rede; os dispositivos realizam um processo de autenticação verificando se a rede é segura e a desejada para receber informação; encriptação da informação, garantido que em casos de esta ser intercetada não pode ser lida. Contudo existe um *trade off* associado na obtenção destes elementos de segurança. Relativamente à encriptação esta não deve causar *overhead* nas mensagens de modo a impedir que o volume de dados seja alargado. O custo energético é também um fator a ter em consideração e como tal as operações relacionadas com segurança não devem consumir uma quantidade significativa de energia. Por último, pode também conter a capacidade de realizar *upgrades* na segurança, ou seja, caso uma falha no sistema de segurança seja descoberta esta pode ser corrigida recorrendo, assim, a um *upgrade over the air* na rede [12].

2.3.5 Capacidade de Rede

Atualmente grande parte das redes LPWA existentes não possuem constrangimentos relativamente à capacidade da rede. Porém, se as estimativas realizadas revelarem-se assertivas, futuramente milhares de dispositivos vão estar conectadas entre si, colocando em causa a capacidade da rede, relevando-se assim um fator fulcral no dimensionamento da mesma. A capacidade não esta apenas relacionada com quantidade de dispositivos, conectados entre si, mas também com o tamanho das mensagens, tempo das transmissões, frequência das transmissões e interferências.

Abordando do ponto de vista das redes telefónicas, a capacidade é vista como a quantidade de dados enviada por unidade de espectro de rádio (bits/Hz) pela operadora. Para uma rede LPWA, numa primeira análise, os objetivos são semelhantes ao das redes telefónicas, porém existe alguma divergência em certos pontos.

As mensagens enviadas por parte dos dispositivos presentes na rede são geralmente curtas. Analisando um sensor de estacionamento, este necessita apenas de 1 bit que indica se o local está livre ou ocupado. Mensagens com o intuito de atualizar a localização de um determinado objeto tem grandes probabilidades de consumir bastantes recursos da rede. Examinando como exemplo um veículo em que a sua posição é atualizada, mas a informação é apenas enviada uma vez ao dia, é capaz de enviar mil vezes mais dados do que o utilizador necessita. O formato das mensagens desempenha um fator importante na rede podendo aumentar a capacidade desta em dez vezes mais.

Os utilizadores de telemóveis tendem aceder aos dispositivos em tempos não pré-definidos, isto é, o envio de uma mensagem ou a realização de uma chamada pode ser a qualquer altura (*random time*). O dispositivo passa, assim, por uma fase de *random acess* (acesso não previsto),

para iniciar comunicações com a rede, sendo posteriormente providenciados recursos dedicados durante a sessão. Este método é pouco eficiente num ambiente constituído por dispositivos que requerem pouco consumo energético. A possibilidade de diversos dispositivos acederem aos recursos da rede ou o envio de mensagens em simultâneo, pode causar o colapso da mesma sendo posteriormente necessário as transmissões serem realizadas novamente e possivelmente recorrer a um *reset* na rede. Uma solução passa por envios periódicos dos dados, isto é, os dispositivos sabem quando devem enviar a informação sendo atribuído um intervalo de tempo entre as mensagens, recorrendo assim a uma metodologia capaz de obter uma eficiência três vezes superior.

As redes telefónicas atuais possuem o seu próprio espectro e como tal não correm o risco de interferência por parte de outras. Contudo, no caso das LPWAN o espectro não dispõe de licença (em algumas tecnologias) existindo, assim, a possibilidade de interferências por parte de outras redes LPWA que utilizem a mesma tecnologia ou até mesmo por outros utilizadores. Este fenómeno, posteriormente tende a causar adversidades, visto que ao longo do tempo mais redes LPWA são implementadas. Para tal, são utilizadas técnicas como *frequency hopping* e *message acknolwage* com o intuito de fornecer às redes a capacidade de se adaptarem a ambientes onde existam interferências [13].

2.3.6 Custos dos Dispositivos

Nas diversas tecnologias que são abordadas futuramente no próximo capítulo, normalmente o responsável pelo design da rede possui custos associados à mesma tal como os dispositivos e o acesso à rede, em algumas das tecnologias (semelhante as redes telefónicas). Diversas aplicações que recorrem a tecnologias LPWAN, tendem a utilizar um número elevado de nós o que requer um custo reduzidos destes, o que é um fator importante ao longo do design da rede. Comparativamente às tecnologias que apresentam curto alcance ou tem como base topologias mesh, as redes LPWAN possuem gateways a longas distâncias em relação aos nós proporcionando uma redução de infraestruturas e consequentemente de custos. Quanto à manutenção destes diapositivos, os custos associados são o menor quanto possível sendo toda a complexidade realizada por parte dos gateways e do servidor de rede.

2.4 Tecnologias LPWAN

Nesta secção, pretende-se uma avaliação das diversas tecnologias existentes no mercado. É realizada uma breve análise de três tecnologias LPWAN, incluindo LoRaWAN, concluindo com uma comparação entre as três onde é abordada a escolha da tecnologia optada na presente dissertação.

2.4.1 LoRaWAN

A tecnologia LoRaWAN (*Long Range Wide Area Network*) desenvolvida pela Semtech e LoRa Alliance, surge em 2015 como um meio de comunicação wireless que recorre a bandas de frequências não licenciadas Sub-GHz. Numa primeira instância, torna-se fulcral estabelecer uma distinção entre LoRa e LoRaWAN de modo a compreender como estas se encontram interligadas e quais as suas funções. LoRa é a camada física e como tal é responsável pela modulação do sinal que torna possível o longo alcance e uma baixa probabilidade de interferências, enquanto o LoRaWAN define o protocolo de comunicação ou seja é a camada MAC com suporte *low power*, longo alcance e com uma capacidade de rede elevada. LoRa baseia-se na técnica de modulação *Chirp Spread Spectrum* (CSS), que permite obter longos alcances (15 km) auxiliado de um consumo energético baixo quando comparado a tecnologia wireless que recorrem a técnicas de modulação como *Frequency Shift Key* (FSK). A técnica CSS, é semelhante à utilizada por parte da tecnologia Sigfox, *Ultra Narrow Band* (UNB), apresentando também uma robustez elevada relativamente a perdas de sinais e contra interferências.

Relativamente à arquitetura utilizada, é baseada numa topologia em estrela onde a informação pode ser transmitida de forma bidirecional ou seja de nós para gateways e no sentido oposto. A camada MAC estabelece para cada dispositivo uma classe (A, B ou C). Estas possuem diferentes funcionalidades e características de modo a permitir o enquadramento do LoRaWAN nas mais diversas aplicações. O envio de mensagens *uplink* é realizado de forma assíncrona não existindo qualquer tipo de sincronização entre os nós e gateway, isto é o envio de informação é realizado quando o dispositivo se encontra preparado para executar tal processo. Este método de operação permite obter menores consumos energéticos quando comparado sistema que recorrem a tecnologias móveis em que os dispositivos enviam informação de forma regular e sincronizada. O envio de mensagens *downlink* é geralmente realizado após a receção de uma mensagem *uplink* transmitida por parte do *nó*, sendo que este processo encontra-se dependente da classe do *nó*. O LoRaWAN permite também ao utilizador optar pela confirmação da receção de mensagens, sendo que este processo varia consoante a classe atribuindo assim maior fiabilidade e flexibilidade à tecnologia em causa [14].

No que toca à segurança, esta tecnologia difere das restantes em análise nesta secção. O LoRaWAN recorre à utilização de diferentes chaves consoante o método de ativação do dispositivo (OTAA ou ABP), o que lhe confere uma elevada flexibilidade, apresentando uma fiabilidade semelhante ao Ingenu RPMA que, tal como o LoRaWAN, possui uma sólida implementação a nível de segurança da informação transmitida. Esta tecnologia é abordada com maior detalhe, camada física e camada MAC, na seguinte secção devido à sua utilização durante o desenvolvimento do projeto em questão.

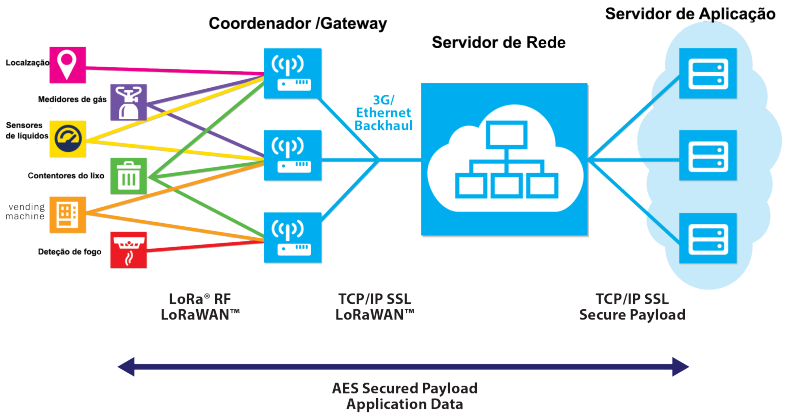


Figura 2.2: Arquitetura de rede utilizando tecnologia LoRaWAN. A imagem representa, de forma sucinta, as três camadas da rede e de que modo comunicam entre si. Entre nós e *gateway* é utilizado tecnologia LoRa (Adaptado de [14]).

2.4.2 SigFox

A empresa Sigfox surge em 2009 criando colaborações com diversas operadoras em diferentes partes do globo de modo a fornecer soluções Low Power Wide Area através da criação de redes com essas mesmas características. A Sigfox pretende fornecer os mesmos serviços disponíveis pela redes moveis, porém tem como alvo dispositivos com baixo consumo energético e consequentemente a pouca necessidade de estes realizar processamentos complexos. Atualmente, a empresa possui redes implementadas em quarenta e cinco países diferentes tendo como perspetiva futura tornar-se no líder mundial do sector LPWAN.

Os dispositivos utilizados por parte da Sigfox, recorrem à utilização da modulação *Ultra Narrow Band* a qual permite obter um *link budget* alto e em simultâneo uma largura de banda pequena. Devido a esta última característica mencionada, o ruído ou interferência possíveis são bastantes reduzidos permitindo também um elevado número de dispositivos por largura de banda. Contudo, a utilização desta técnica conduz à obtenção de *data rates* baixos e um consequente aumento no tempo necessário durante a transmissão de mensagens e consequentemente do funcionamento do modulo de radio.

As redes Sigfox possuem uma topologia em estrela, como se encontra representado na Figura 2.3, oferecendo uma cobertura e propagação semelhantes às redes telefónicas devido à sua modulação, sendo apenas necessário poucos gateways para obter a cobertura de uma grande área (como a de uma cidade). Os dados são transmitidos a partir de dispositivos (nós sensores ou *edge of the network devices*) para um gateway que posteriormente envia os mesmo para o servidor de rede Sigfox recorrendo a 3G/4G ou Ethernet. Por último, a informação é armazenada no servidor da Sigfox, ou reencaminhada para a aplicação do cliente.

Relativamente à segurança a Sigfox, esta estabelece diversas camadas começando no nó e terminando apenas na aplicação destinada ao cliente. De modo a ser realizada a transmissão

de informação e comunicação com o servidor de rede, cada mensagem necessita de possuir uma chave de autenticação única que posteriormente é utilizada para realizar a encriptação da mensagem a ser transmitida [15].

Figura 2.3: A imagem representa a estrutura de um rede Sigfox semelhante à LoRaWAN. Nesta Figura é de notar as camadas distintas e a segurança existente entre cada uma destas (Adaptado de [15]).

2.4.3 Ingenu RPMA

Ao contrário das tecnologias analisadas até este ponto, a Ingenu opera utilizando a banda de frequências 2.4 GHz (Wi-Fi e Bluetooth) em vez de recorrer à banda Sub-GHz (769 – 935 MHz, 315 MHz e 468 MHz) permitindo assim excluir preocupações relacionadas com a localização geográfica onde a rede é implementada conduzindo a uma maior flexibilidade. Devido à operação na banda de 2.4 GHz, torna-se possível obter uma maior largura de banda bem como um aumento na taxa de transferência e capacidade da rede comparativamente as tecnologias que recorrem as bandas Sub-GHz.

A Ingenu recorre à tecnologia *Random Phase Multiple Access* (RPMA), desenvolvida pela empresa em questão e que funciona como camada física e camada MAC, com o intuito de conseguir alcançar a máxima otimização da utilização da banda 2.4 GHz. Possui características principais como a cobertura extensa na ordem dos $250km^2$ auxiliado de um gatewy *low cost* e a enorme capacidade de rede, respeitando em simultâneo a métrica de baixo consumo energético.

Semelhante às arquiteturas já analisadas anteriormente, a Ingenu recorre a uma topologia em estrela, representada na Figura 2.4. Permite comunicação bidirecional, ou seja, possui a capacidade de transmitir mensagens de *downlink* e *uplink*, sendo totalmente necessária a transmissão de dados entre o servidor de rede e os nós pelo facto que esta tecnologia confirmar a receção de todas as mensagens *uplink* através da utilização de *acknowledges*, atribuindo deste modo maior viabilidade à tecnologia [16].

No que toca a segurança, a Ingenu RPMA oferece garantias através de seis procedimentos com a capacidade de fornecer a fiabilidade das mensagens, bem como toda a integridade da rede. Recorre inicialmente a um método comum, que passa pela utilização de chaves únicas de modo a encriptar a informação a ser transmitida, porém estes mesmo dados podem ser capturados e enviados provocando mais tarde comportamentos indesejados na aplicação destinada.

Deste modo, o segundo método de proteção garante a integridade da mensagem e que esta não é válida para ser retransmitida, além deste segundo método de segurança a tecnologia possui também *mutual authentication*, isto é, os nós estão associados a uma rede especifica e como tal são autentificados de modo a garantir a sua fiabilidade, impedindo que nós "desconhecidos" à rede estabeleçam conexão com a mesma. A possibilidade de *multi-cast* (enviar uma mensagem para um determinado grupo de dispositivos) por parte desta tecnologia é também protegida através de autenticação com o objetivo de verificar quais os *nós* destinados à informação e se estes são viáveis. Os restantes dois processos de segurança, consistem em preservar o anonimato dos dispositivos, impedindo o acesso de fontes "inseguras" aos identificadores dos *nós* e possibilitar *upgrades* de firmware de forma segura, sendo que através destes são implementadas novas medidas de proteção [17].

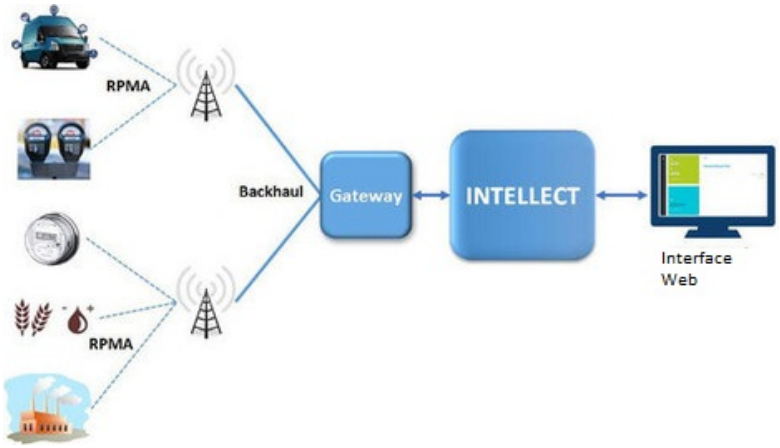


Figura 2.4: Rede Ingenu com os diferentes componentes da tecnologia (Adaptado de [17]).

2.4.4 Escolha da tecnologia LPWAN

Após uma breve análise das três tecnologias LPWAN apresentadas anteriormente, é possível retirar as características fundamentais de cada uma e assim realizar uma comparação exaustiva, com intuito de desenvolver uma melhor compreensão relativamente à tecnologia adotada, LoRaWAN, para este trabalho de dissertação.

No que toca à banda de frequências utilizadas, apenas difere a tecnologia Ingenu RPMA das restantes duas, LoRaWAN e Sigfox. Enquanto, a Ingenu opera numa banda licenciada (2.4 GHZ), o que facilita a introdução destas redes em diversas partes do globo, as tecnologias LoRaWAN e Sigfox recorrem a bandas não licenciadas e como tal encontram-se restringidas com 1% duty-cycle (tempo que o dispositivo pode estar ligado) em mensagens de *uplink*, afetando assim a latência e as transmissões *downlink*. Devido à utilização da banda 2.4 GHz, a tecnologia desenvolvida pela Ingenu consegue disponibilizar valores de *data rate*, 78 kbps uplink e 19.5 kbps *downlink*, superiores aos apresentados por parte do LoRaWAN, 37.5 kbps *downlink* e *uplink*, e

Sigfox, 100 bps *uplink* e 600 bps *downlink*, verificando-se também valores superiores de *data rate* pela tecnologia LoRaWAN em relação à Sigfox. Relativamente ao tamanho máximo de *payload*, o Ingenum RPMA, novamente, revela-se superior possuindo um *payload length* de 10 kB, seguindo-se LoRaWAN com 250 Bytes, sendo o Sigfox a tecnologia com um tamanho menor entre as três com 12 Bytes *uplink* e 8 Bytes *downlink*.

Porém, a utilização da banda 2.4 GHz por parte da Ingenu apresenta algumas desvantagens comparativamente as restantes tecnologias analisadas. Devido à utilização da frequência mencionada, a possibilidade de interferências é elevada bem como a existência de perdas na propagação do sinal o que não se verifica nas tecnologias LoRaWAN e Sigfox que recorrem a técnicas de modulação com a capacidade de evitar ruídos e interferências de fatores externos à rede.

Restantes fatores que influenciam na escolha da tecnologia a adotar e que fazem parte das características principais de uma tecnologia LPWAN são o baixo consumo energético e o alcance máximo. As três tecnologias apresentam maior alcance em ambientes *outdoor* e como tal serão considerados esses valores. O LoRaWAN permite atingir 15 km conseguindo em simultâneo preservar a bateria do dispositivo de modo a que esta dure em média mais de 10 anos (consoante o tempo de funcionamento do mesmo), em semelhança à tecnologia Sigfox que possui a capacidade de oferecer um alcance de 50 km revelando-se superior relativamente as restantes LoRaWAN e Ingenu, sendo que esta última atinge 15 km. Deste modo através da utilização de um gateway apenas, o Sigfox permite cobrir áreas extensas na qual estão inseridos diversos nós levando assim à necessidade de um menor número de gateways comparativamente as restantes tecnologias. No que concerne consumo energético, o Sigfox e LoRaWAN possuem os consumos mais baixos, contribuindo para este fator a utilização das frequências Sub-Ghz não licenciadas (1% duty cycle), enquanto a tecnologia Ingenu RPMA requer um maior consumo energético devido à necessidade de custos significativos durante todo o processamento.

A instalação da rede é também um fator a ter em conta na decisão por qual tecnologia optar bem como os seus custos associados. O LoRaWAN demonstra possuir a maior flexibilidade relativamente à introdução de nós numa rede, possuindo dois métodos distintos de ativação destes dispositivos: OTAA e ABP. A possibilidade de evitar a configuração do nó manualmente através do OTAA, oferece menor complexidade quando comparado ao método utilizado por parte da Sigfox e Ingenu, onde existe a necessidade de realizar configurações prévias no *nó*. Relativamente ao custo dos componentes, as três tecnologias apresentam valores semelhantes. Porém, a LoRaWAN possui limitações quanto aos produtos disponíveis, devido ao facto de atualmente existir apenas um fabricante de transceivers RF integrados LoRa, Semtech, sendo que as restantes tecnologias possuem uma oferta mais alargada quanto aos chips de modulação utilizados.

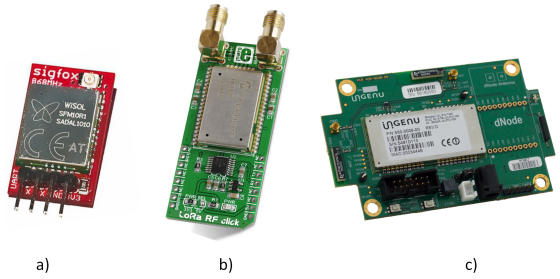
Por outro lado, existem custos de serviços associados na implementação de uma rede LPWAN que variam nestas três tecnologias. No caso da Sigfox, é necessário realizar o pagamento de

serviços cloud, de modo a contemplar as três camadas da rede (nós, gateways e servidor de rede) e obter o total funcionamento da rede, principalmente *uplinks*, a mesma situação é verificada na Ingenu, sendo que esta possui um carácter mais focado em empresa oferecendo serviços completos implementado a rede desde raiz. Nesta temática, a LoRaWAN revela-se mais vantajosa existindo plataformas gratuitas que limitam apenas o número de mensagens transmitidas, mas que fornecem os serviços básicos para a construção de uma rede excluindo a necessidade de requisitar a implementação a empresas.

Em suma, cada tecnologia possui as suas vantagens e desvantagens tal como características únicas que as distinguem entre si bem como limitações que impedem ou restringem a sua utilização no âmbito desta dissertação. A necessidade de "criar" uma rede desde raiz incluindo os componentes associados, evitando custos adicionais, revela-se um ponto forte por parte do LoRaWAN demonstrando possuir flexibilidade em comparação com as restantes tecnologias. Relativamente ao consumo energético, ambos LoRaWAN e Sigfox apresentam consumos reduzidos sendo o Ingenu RPMA o que revela a menor longevidade de nós. A possibilidade de interferências, é também um fator a ter em consideração, o qual pode resultar a perda ou alterações na transmissão de pacotes. A utilização de bandas 2.4 GHz por parte do Ingenu, eleva a possibilidade destas situações surgirem durante o funcionamento da rede e como tal colocam em risco aplicações onde a informação contida nas mensagens possui conteúdo crítico. Apesar do Ingenu apresentar o maior *payload length* das três tecnologias, este ponto não se revela fulcral ou necessário no âmbito da dissertação em questão. Assim, a exclusão da tecnologia Ingenua RPMA justifica-se pelos consumos energéticos superiores, bem como, as possíveis interferências, sendo esta uma tecnologia indicada para aplicações que requerem o envio de grandes quantidades de informação. Apesar da semelhança entre as tecnologias Sigfox e LoRaWAN, os serviços pagos por parte da Sigfox excluem esta tecnologia bem como o limite imposto pela mesma nas quantidades diárias máximas de mensagens transmitidas. Deste modo, o LoRaWAN foi a tecnologia indicada a adotar neste projeto oferecendo liberdade nas três camadas *nó*, gateway e servidor de rede, aliado de um baixo consumo energético e indicado para aplicações com uma troca de mensagens mínima para com o servidor, adequada para a criação da rede desejada. Uma abordagem em maior detalhe relativamente à tecnologia LoRaWAN, encontra-se localizado no seguinte subcapítulo.

Tabela 2.1: Tecnologias LPWAN.

	LoRa	**Sigfox**	**Ingenu RPMA**
Alcance	15 km (Rural), 5 km (Urbano)	50 km (Rural), 10 km (Urbano)	15 km (Urbano)
Banda	Sub-Ghz	Sub-Ghz	2.4 Ghz
Modulação	*Spread Spectrum*	*Ultra Narrow Band*	RPMA
Tamanho do Payload	250 bytes (consoante SF)	12 bytes (*Uplink*), 8 bytes (*Downlink*)	10 kbytes
Encriptação	AES 128b	Não suportado	16B hash, AES 256b
Data Rate	37.5 kbps (*Uplink* e *Downlink*)	100 bps (*Uplink*), 600 bps (*Downlink*)	78 kbps (*Uplink*), 19.5 kbps (*Downlink*)
Preço Nós	3-5€	<2€	-



a) b) c)

Figura 2.5: Os dispositivos de cada uma das tecnologias estudadas: a) Sigfox; b)LoRa; c)Ingenu RPMA; [18] [19] [20].

2.5 LoRa e LoRaWAN

Após uma breve introdução da tecnologia LoRa/LoRaWAN através da exploração das vantagens e desvantagens desta em relação a outras tecnologias, torna-se relevante e necessário detalhar toda a camada física e de protocolo sendo que esta é a tecnologia adotada no desenvolvimento do projeto em questão. Este capítulo destina-se, assim, a uma abordagem pormenorizada do funcionamento desta tecnologia, analisando o seu comportamento a nível de segurança, de que forma é realizada a gestão da rede, o formato pré-definido das mensagens e quais os parâmetros que influenciam o alcance e possíveis interferências na rede. Por último, e com o intuito de fornecer não só uma análise no que toca as vantagens da tecnologia, são também abordadas as vulnerabilidades a nível de segurança explorando possíveis ataques à rede e de como alguns destes podem ser contornados. Ao longo de toda a dissertação, foi utilizada a versão 1.0 do LoRaWAN devido ao elevado suporte disponibilizado (hardware e software), porém atualmente

encontra-se disponível a versão 1.1, que apresenta modificações significativas no que toca à segurança.

2.5.1 LoRa PHY

Como anteriormente, LoRa é a camada física (LoRa PHY) utilizada para comunicações de longo alcance, devido à sua modulação. Os circuitos integrados COTS, que implementam LoRa PHY, disponíveis são apenas desenvolvidos pela Semtech. Ao contrário de diversos sistema wireless que recorrem a *Frequency Shift Key* (FSK) com o intuito de obter uma melhor otimização a nível de bateria, a tecnologia LoRa baseia-se na modulação *Chirp Spread Spectrum* (CSS), a partir da qual consegue atingir a mesma característica *low power* da modulação FSK, mas obtendo um maior alcance.

A vantagem apresentada pela LoRa, é a capacidade de permitir comunicações a longas distâncias consoante o ambiente em que se enquadra, sendo que através da utilização de um gateway apenas é possível abranger áreas com dezenas de quilómetros, dependendo da existência de obstáculos na localização no qual são inseridos. Porém, a tecnologia LoRa apresenta um *link budget* mais elevado que as restantes redes atuais e como tal é um fator fulcral na determinação do alcance máximo da rede.

A utilização da tecnologia em questão, requer que a comunicação entre nós e gateways seja realizada recorrendo a *frequency channels* (gateways possuem canais com diferentes frequências) diferentes bem como *data rates* (variam entre 0,3 kbps e 20 kbps). A decisão pelo *data rate* indicado, surge de um *trade-off* entre o alcance e a duração da mensagem, sendo que dispositivos com *data rates* diferentes não interferem entre si.

2.5.1.1 Modulação LoRa

O processo de modulação tem como objetivo modificar um sinal ou dados a transmitir, de modo a produzir um sinal apropriado ao meio de transmissão. As técnicas de modulação permitem alterar as características do sinal, que se pretende transmitir, com o intuito de adapta-lo às características do canal. Nas tecnologias LPWAN existem dois tipos de modulações standard: *Ultra Narrow Band* (UNB) e *Spread Spectrum Modulation* (SSM). De forma breve, relativamente ás UNB estas oferecem um alcance superior nas comunicações, enquanto as SSM apresentam maior robustez a interferências.

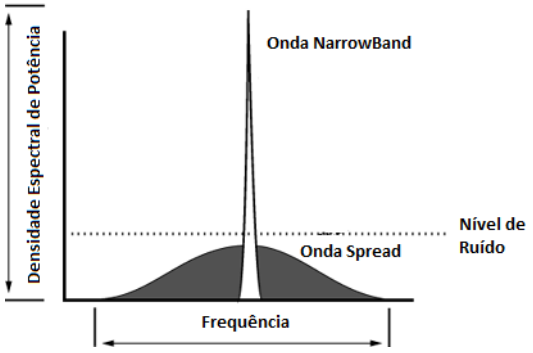


Figura 2.6: Dois tipos de onda de cada uma das modelações: *Spread Spectrum* e *Ultra Narrow Band*. A partir da Figura é possível verificar uma das características principais da modelação SS, a capacidade de emitir abaixo do nível de ruído contrariamente ao UNB. (Adaptado de [21])

A modulação *Spread Spectrum* difunde o sinal, de acordo com uma sequência, na totalidade da largura de banda do canal gerando, assim, um sinal abaixo do nível de ruído, como presente na Figura 2.6. O recetor recebe o sinal codificado tornando-se necessário proceder à sua descodificação (demodulação), sendo esta realizada através do inverso da mesma sequência de propagação obtendo, desta forma, a informação do emissor. A tecnologia LoRa, recorre à técnica *Spread Spectrum* através da geração de um sinal *chirp* que varia a frequência do mesmo continuamente.

O LoRa, tem assim como base de modulação a técnica *Chrip Spread Spectrum*. O *Chrip* consiste num sinal sinusoidal com a capacidade de aumentar (*Chrip-Up*) ou diminuir (*Chrip-Down*) a frequência, ao longo do tempo, de acordo com um *chirp rate* (número de vezes que a frequência muda). A imagem 2.7, representa dois exemplo de sinais *chirp up* e *chirp down*, onde é possível verificar a mudança da frequência com o decorrer do tempo [22].

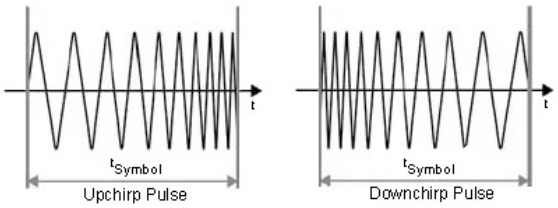


Figura 2.7: Representação de dois sinais: *Chrip Up* e *Chrip Down* [23].

2.5.2 LoRaWAN Camada MAC

LoRaWAN é o protocolo de rede ou camada MAC desenvolvido pela LoRa-Alliance, do qual fazem parte empresas e universidades. Este protocolo baseia-se na camada física LoRa desenvolvida pela Semtech sendo responsável pela segurança da rede, do consumo energético dos *nós*, pela qualidade do serviço e determinar qual a melhor maneira de respeitar os requisitos das mais diversas aplicações possíveis. O LoRaWAN, é assim considerado um protocolo LPWAN com

características distintas, como tem vindo a ser abordado ao longo deste documento e em maior detalhe no presente capítulo, ideal no âmbito das aplicações IoT. A Figura 2.8 representa a *stack* LoRaWAN, no qual estão realçadas as três camadas diferentes sendo que ao longo deste capítulo é realizada uma abordagem pormenorizada à camada MAC ou seja protocolo LoRaWAN.

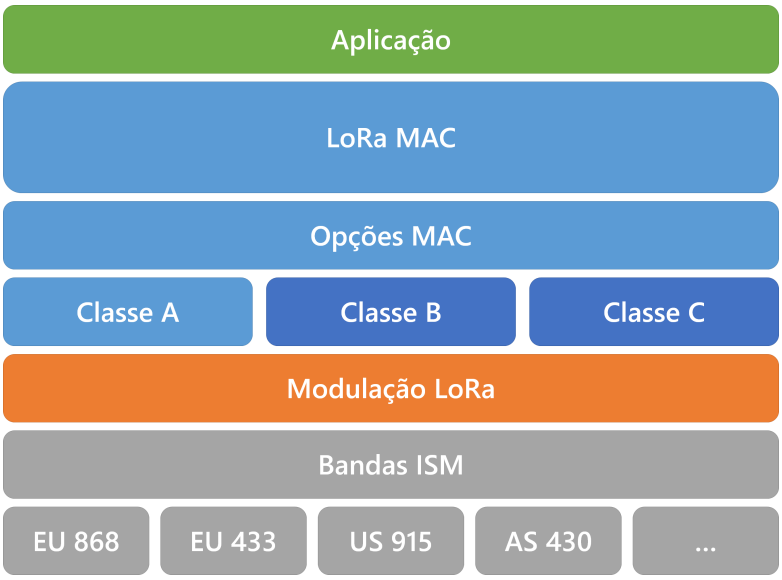


Figura 2.8: Representação da stack LoRaWAN atribuindo maior relevo as camadas LoRa MAC, Modulação LoRa e Aplicação. Esta stack encontra-se presente em todos os nós e gateways que recorrem a esta tecnologia (Adaptado de [24]).

2.5.3 Classes LoRaWAN

Como já foi referido anteriormente, os *nós* de uma rede LoRa podem possuir uma das três classes distintas: classe A, classe B ou classe C. As três classes tem como objetivo satisfazer diferentes aplicações tal como cumprir os requisitos impostos com o intuito de otimizar os dispositivos presentes na rede. Estas classes apresentam diferentes *trade-offs* que influenciam fatores como o tempo de vida do dispositivo ou a latência na receção de mensagens, o que conduz a uma grande importância na decisão da classe pela qual optar podendo esta escolha influenciar todo o comportamento da rede. Assim, este subcapítulo tem como objetivo detalhar cada uma das classes disponíveis por parte da tecnologia LoRa, de modo a elaborar uma melhor compreensão de como estas podem ser aplicadas e quais as caraterísticas que as diferenciam.

2.5.3.1 Classe A

Os dispositivos do tipo classe A, apresentam-se como sendo os menores em termos de consumo energético e com a capacidade de realizar uma comunicação bidirecional com o servidor de rede. As mensagens enviadas por parte dos dispositivos, *uplink*, são seguidas de duas janelas curtas de receção, tal como se encontra representado na Figura 2.9, o envio de informação é realizado

aleatoriamente sendo uma decisão dependente apenas do nó. Tal como foi referido, o consumo por parte dos dispositivos de classe A é o menor comparativamente as restantes classes, sendo deste modo ideal para aplicações em que seja necessário o envio de mensagens *uplinks*, como por exemplo sensores para o controlo de temperaturas. Para situações na qual existe a necessidade de mensagens *downlink*, estas devem ser enviadas após a receção do *payload* proveniente do nó de modo a conseguir ser intercetada por uma das janelas de receção, caso a mensagem de *downlink* falhe o envio desta deverá ser transmitida na próxima comunicação para com o gateway [25].

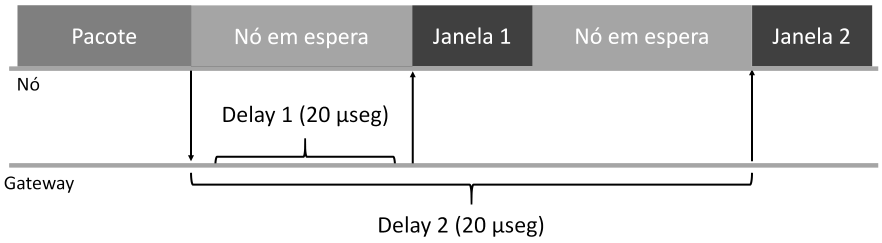


Figura 2.9: Classe A de nós (Adaptado de [25]).

2.5.3.2 Classe B

Semelhante aos dispositivos de classe A, a classe B permite também uma comunicação bidirecional, porém o número de janelas de receção é superior. A receção por parte do nó, é "agendada" contrariamente aos de classe A, sendo esta funcionalidade obtida através da emissão de um *beacon*, sincronizado, por parte do gateway com o objetivo de determinar em que altura o nó se encontra preparado para receber informação. Aplicações no qual seja necessário preservar a longevidade do dispositivo e em simultâneo atuar consoante valores recebidos, como realização da leitura de um sensor onde são necessárias amostras periódicas, a classe B revela-se indicada para dispositivos enquadrados nesse tipo de aplicações [25].

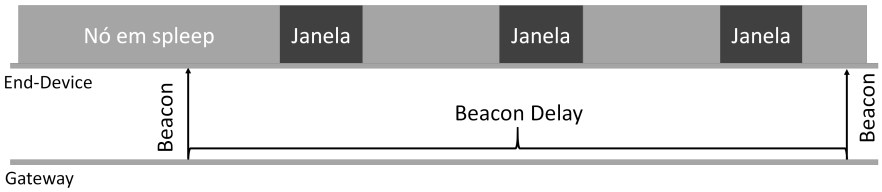


Figura 2.10: Classe B de nós (Adaptado de [25]).

2.5.3.3 Classe C

Contrariamente às duas últimas classes analisadas, a classe C mantém a janela de receção na maior parte do tempo aberta sendo apenas fechada nos momentos em que se encontra a transmitir. Contudo, o consumo energético aumenta e a longevidade da bateria diminui em comparação as restantes classes oferecendo por outro lado uma baixa latência na comunicação

entre o servidor de rede e o dispositivo de classe C. A utilização desta classe, destina-se principalmente a aplicações onde seja necessário uma comunicação *downlink* constante com os dispositivos como por exemplo *car tracking* ou *industrial control* [25].

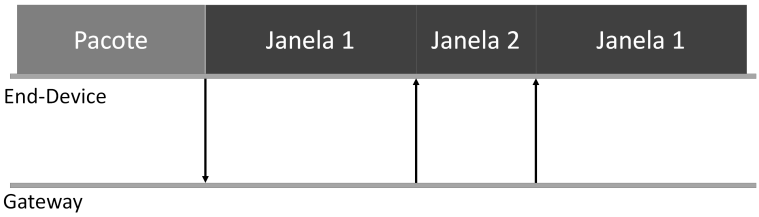


Figura 2.11: Classe C de nós (Adaptado de [25]).

2.5.4 Parâmetros LoRaWAN

Um dispositivo incorporado com tecnologia LoRa, possui a capacidade de ser configurado recorrendo a parâmetros como *Transmission Power* (TP), *Carrier Frequency* (CF), *Spreading Factor* (SF), *Bandwidth* (BW) e *Coding Rate* (CR) de modo a obter um desempenho e consumos energéticos adequados. A seleção destes parâmetros possui um grande impacto na performance da rede, influenciando significativamente o alcance, a diminuição do risco de possíveis ruídos ou interferências e como já mencionado o consumo energético que coloca em causa a longevidade dos nós. O objetivo principal, é assim conseguir obter um equilíbrio entre a performance das comunicação e os consumos energéticos, parâmetros como TP possuem grande influência na bateria do nó, bem como os parâmetros SF e BW que influenciam o *air time* na transmissão de pacotes. Assim, este subcapítulo detalha quais os principais e únicos parâmetros possíveis de configurar, mencionando a importância destes na aplicação e nos dispositivos na rede em que se enquadra.

2.5.4.1 *Transmission Power* (TP)

Um dos parâmetros mais comuns em diversas tecnologias wireless é a potencia da transmissão. Este encontra-se também presente na tecnologia LoRa. O TP é ajustável entre -4dBM e 20dBM (com incrementos de 1dB), sendo que quanto maior for o valor associado maior será o consumo energético necessário para transmitir um pacote, porém proporciona um maior alcance [26].

2.5.4.2 *Carrier Frequency* (CF)

O *Carrier Frequency* é responsável por determinar qual a frequência utilizada pelo o modulo LoRa durante a transmissão de pacotes. Esta pode obter valores entre 137 MHz e 1020 MHZ através de incrementos de 61 MHZ. Porém, a tecnologia LoRa limita o intervalo entre 860 MHz e 1020 MHz mantendo o mesmo valor de incremento [26].

2.5.4.3 *Coding Rate* (CR)

O *Coding Rate* é o método de *Fowardig Error Correction* (técnica de verificação e controlo de erros na transmissão de dados) utilizada por parte da tecnologia LoRa com o intuito de fornecer proteção contra interferências, conduzindo assim a uma maior fiabilidade. Ao CR podem ser atribuídos quatro valores diferentes: 4/5, 4/6, 4/7, 4/8. Quanto maior for o valor selecionado maior é a proteção oferecida, porém o *time on air* é por sua vez mais longo [26].

2.5.4.4 *Spreading Factor* (SF)

O *Spreading Factor* é definido de uma forma breve e simples, como sendo a duração de um *chirp*, conceito que foi abordado no subcapítulo relacionado com a modulação LoRa. A tecnologia em questão utiliza valores de *Spreading Factor* entre 7 e 12, sendo o primeiro o mais rápido e por sua vez aquele que necessita de menor *time on air*, por outro lado o SF12 é o mais lento o que conduz a um maior *time on air*. Consequentemente, quanto maior o SF utilizado maior será a distância entre o nó e o gateway, tal como o inverso.

$$DataRate = SF \cdot BW/2^{SF} \tag{2.2}$$

Equation 2.2: Equação de obtenção do *data rate*.

O *data rate*, o *bandwith* e o *spreading factor*, encontram-se relacionados através da expressão apresentada. A Figura seguinte demonstra os valores obtidos recorrendo a um *bandwidth* de 125 kHz para diferentes valores de SF. A partir dos dados apresentados é possível denotar que a velocidade de transmissão vai diminuindo sempre que o SF é aumentado [27].

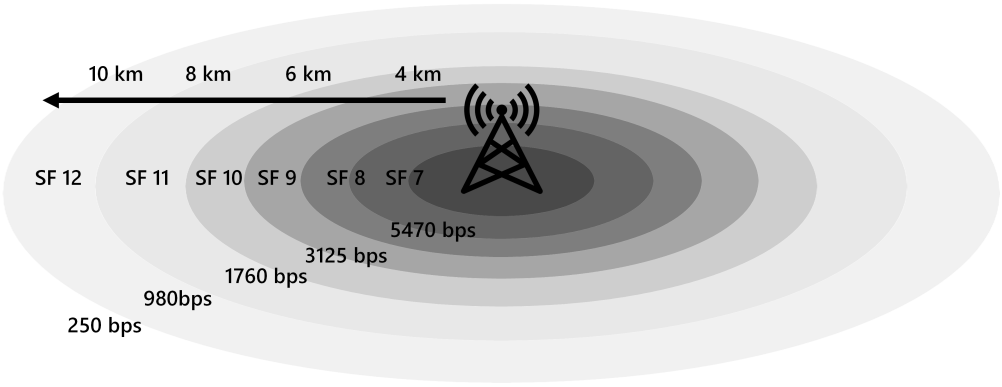


Figura 2.12: A Figura representa os alcances e velocidades das transmissões consoante o SF escolhido (Adaptado de [28]).

2.5.4.5 *Bandwidth* (BW)

O *Bandwidth* ou largura de banda, é o intervalo de frequências na qual a tecnologia LoRa pode transmitir. Quanto mais elevado for a BW maior será o *data rate* e consequentemente conduz a

uma redução no *time on air*, porém o sinal fica suscetível a possíveis ruídos. Na tecnologia LoRa os valores de *bandwidth* usuais variam entre os 125kHz e 500kHz, sendo que valores inferiores requerem a utilização de um cristal mais preciso com o objetivo de evitar problemas relacionados com *clock* [27].

2.5.4.6 *Signal to Noise Ratio* (SNR) e *Received Signal Strength Indicator* (RSSI)

O parâmetro não configurável *Signal to Noise Ratio*, determina a diferença, em decibéis, entre o sinal recebido e a soma de todos os possíveis ruídos que afetam o sinal. Este parâmetro é apresentado por parte do gateway, tendo como intuito analisar a qualidade do sinal juntamente com o RSSI. Este ultimo permite realizar a medição da força do sinal, apresentando também os valores em decibéis. O SNR pode variar entre -20dB, onde o sinal esta perto da zona de ruído, sendo que para valores superiores a 5dB o sinal encontra-se em boas condições. Por outro lado, o RSSI possui valores geralmente entre -30 e -120 dBm em que quanto menor o RSSI mais perto este se encontra na zona de ruído [27].

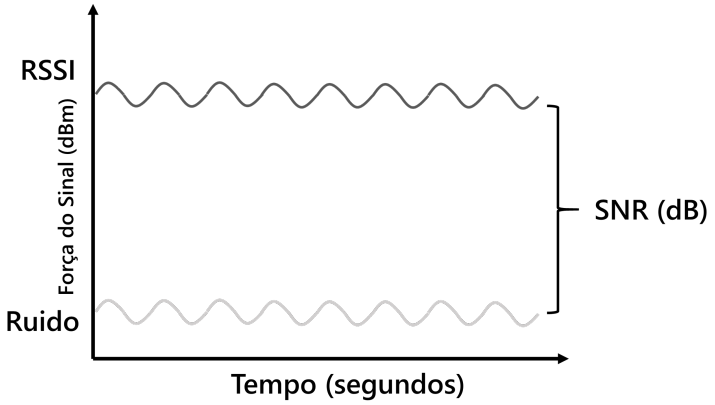


Figura 2.13: Comparação entre o SNR e o RSSI (Adaptado de [29]).

2.5.5 Segurança LoRaWAN

A segurança encontra-se diretamente relacionada com confidencialidade, integridade e acessibilidade, sendo estes aspetos fulcrais numa rede LPWAN. A confidencialidade, permite que a informação possa ser acedida apenas por entidades que possuem autorização para tal impedindo o acesso de indivíduos sem permissão. Quanto à integridade, esta possibilita que o conteúdo não é alterado ou acedido, por indivíduos não autorizados, no processo de envio entre um transmissor e um recetor. Por último, a acessibilidade dos dados e da rede deve ser constante de modo a estar sempre disponível, podendo conter conteúdo fulcral (configurações, necessidade de envio de mensagens) que coloca em risco a segurança da rede ou o próprio funcionamento

correto da mesma. A utilização da modulação *Chirp Spread Spectrum*, contribui também para a segurança da rede devido à sua robustez em relação a ruídos externos. Assim, este capítulo tem como objetivo compreender de que forma a tecnologia LoRaWAN consegue fornecer a segurança necessária de forma a cumprir as propriedades básicas mencionadas.

2.5.6 Chaves LoRaWAN

LoRaWAN recorre a diversas chaves, únicas, de autenticação ao longo do processo de conexão entre nós e gateways, gateways e servidor de rede. Cada uma destas chaves possui um tamanho de 128 bits e são utilizadas na encriptação recorrendo à técnica *Advanced Encryption Standard* (AES), sendo esta um algoritmo *symetric-key*, isto é, a mesma chave é utilizada na encriptação e desencriptação dos dados. As chaves utilizadas pelo LoRaWAN permitem assegurar a confidencialidade, como a encriptação, e a integridade, que envolve aspetos como a autenticidade e validação. A *Application Session Key* (AppSKey) e *Network Session Key* (NwkSKey), revelam-se fulcrais para obter confidencialidade e integridade, estando presentes em todos os *nós* da rede. Para inserção de nós através do método *Over The Air Activation* (OTAA), estas duas chaves são derivadas a partir da *Application Key* (AppKey), por outro lado recorrendo ao método *Activations By Personalization* (ABP) este processo de derivação é excluído sendo apenas necessário introduzir manualmente as chaves AppSKey e NwkSKey [30].

2.5.6.1 Application session key (AppSKey)

Uma chave única presente em cada um dos nós da rede, nunca sendo transmitida *over the air*, mas apenas partilhada com a aplicação presente no servidor. Tem como objetivo encriptar a secção do *payload* destinada à aplicação com o propósito de garantir a confidencialidade do mesmo e permitir segurança *end-to-end* entre o *nó* e a aplicação. Na Figura 2.14, é realçado, a azul, a ligação entre os dois componentes mencionados [25].

2.5.6.2 Network session key (NwkSKey)

Tal como a AppSkey, a NwkSKey é uma chave única presente em cada um dos nós da rede e nunca é transmitida *over the air*. Esta chave é utilizada no cálculo e verificação do MIC de cada mensagem com o objetivo de garantir a integridade da mesma, sendo também usada na encriptação/desencriptação na secção MAC do *payload*. A chave é apenas partilhada com o servidor da rede como se encontra representado pela seta vermelha na Figura 2.14 [25].

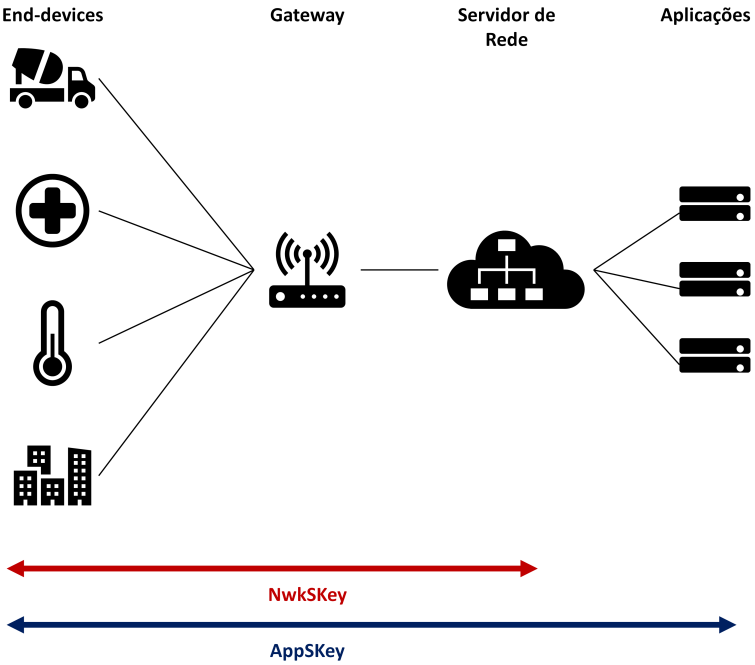


Figura 2.14: Encriptação LoRaWAN (Adaptado de [25]).

2.5.6.3 Application key (AppKey)

É também uma chave única presente nos nós, tal como as mencionadas anteriormente, partilhada entre o servidor da rede e o nó. Durante o procedimento de ativação OTAA esta chave é utilizada na geração das chaves NwkSKey e AppSKey para o dispositivo em especifico apenas. A AppKey é criada pelo utilizador da rede ou gerada aleatoriamente pelo servidor da mesma e possui uma grande importância no âmbito da segurança, devido à responsabilidade pela geração das chaves já mencionadas (acesso a esta chave compromete as restantes) [25].

2.5.7 Métodos de inserção de nós na rede

Tal como já mencionado no presente subcapítulo, um dos componentes presentes numa rede LPWAN são os nós que geralmente incorporam sensores com o objetivo de recolher informações do meio. Sendo o LoRa uma tecnologia LPWAN, as redes que recorrem à mesma possuem também nós, dispersados em área extensas, e todos os restantes componentes para o correto funcionamento (gateway e servidor de rede). A partir destas três camadas, torna-se possível a construção de uma rede Low Power Wide Area, porém é necessário ter em consideração de que modo é realizado a introdução de novos dispositivos na rede sendo para o caso em questão os nós sensores. Deste modo, o presente subcapítulo destina-se a analisar de que forma são inseridos os nós numa rede LoRa através da exploração de dois métodos distintos tendo como fundamentos os pontos abordados no subcapítulo anterior.

2.5.7.1 Over The Air Activation (OTAA)

O primeiro método em análise é designado de *Over The Air Activation*, no qual consiste na introdução de um nó na rede sem existir a necessidade de alterar o nó sendo um *deploy* praticamente imediato. O dispositivo, integra uma chave única (Appkey) que é posteriormente utilizada no momento em que nó envia uma mensagem de *join-request* ao servidor da rede. Para esta situação a chave não é utilizada para realizar a encriptação da mensagem, mas sim com o intuito de identificar o dispositivo em questão. A mensagem a enviar deve conter os dois identificadores relativos ao dispositivo: AppEUI, associado ao gestor da rede, e DevEUI, relativo ao dispositivo num ambiente global. Após a validação da Appkey, o servidor responde com uma mensagem *join-accept* (dentro da janela de receção limitada do nó) sendo posteriormente geradas duas novas chaves únicas: uma para proteção da integridade (NwkSKey) e uma segunda chave para a encriptação do *payload* (AppSKey) [25].

2.5.7.2 Activation By Personalisation (ABP)

O segundo método, *Activation By Personalisation* (ABP) é semelhante ao mencionado anteriormente no que toca à encriptação, porém quando o nó é inserido na rede este possui as chaves únicas armazenadas: NwkSkey e AppSKey. Os dispositivos que recorrem a este processo de *deployment*, evitam o procedimento de *join-request* analisado no método OTAA, conduzindo assim a uma troca imediata de mensagens com o servidor. Para uma melhor compreensão relativamente as duas metodologias, as seguintes imagens representam de forma ilustrativa as diferenças e semelhanças entre ambos [25].

2.5.8 Formato de Mensagem LoRaWAN

O protocolo LoRaWAN estabelece diferenças notáveis entre mensagens *downlink* e *uplink*, especificando de que forma estas são transmitidas. As mensagens *uplink* são enviadas por parte dos nós para o servidor de rede sendo as mesmas retransmitidas por parte de um ou mais gateways, uma mensagem de *downlink* é transmitida apenas para um *nó* passando previamente por um gateway. Este subcapítulo, destina-se assim a realizar uma análise relativa ao formato das mensagens referidas, tendo como base o conteúdo presente no documento *LoRaWAN Specifications*.

2.5.8.1 Camada LoRa PHY

O formato de mensagens utilizado pelo LoRa encontra-se dividido em dois tipos diferentes: explicito e implícito. Pacotes que seguem a estrutura do tipo explicito, o qual é utilizado por defeito, possuem no *header* informações relativas ao número de bytes, *coding rate* e se existe a utilização de *Coding Rate Check* (CRC). A seguinte Figura representa o formato da mensagem, distinguindo

três elementos principais, que serão focados nos próximos subcapítulos: *Preamble*, um *header* opcional e o *payload* de dados. O tamanho da mensagem varia consoante a região e os parâmetros utilizados, o tavmanho varia assim entre 55 e 222 bytes [25].

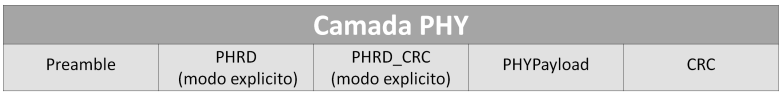


Figura 2.15: Camada rádio PHY.

2.5.8.2 Preamble

O primeiro elemento em análise bem como no formato apresentado é o *Preamble* que tem como função sincronizar o recetor com o número de dados contidos na mensagem. Sendo uma variável programável, é possível alterar o tamanho da mesma com o intuito, por exemplo, diminuir o *duty cycle* do recetor em aplicações onde existe um elevado número de mensagens a receber [25].

2.5.8.3 Header - PHDR

Tal como foi mencionado, existem dois tipos de *headers* distintos consoante o modo de operação escolhido implícito ou explicito. Os próximos dois pontos apresentam em maior detalhe o formato das mensagens utilizados em cada um dos modos referidos.

Modo Explícito

Através da utilização do modo explícito, o qual se encontra selecionado por defeito, é possível incluir informação, relacionado com o *payload* no *header* da mensagem tal como: o tamanho do *payload*, o *forward error correction coding rate* e a utilização de 2 bytes para um CRC opcional relativo ao payload. O *header* em questão, é transmitido com um *Error Correction Code* máximo (4/8), possuindo também um CRC associado de modo a possibilitar o recetor a descartar *headers* inválidos [25].

Modo Implícito

O modo implícito destina-se a aplicações na qual o *coding rate* e CRC do *payload* são fixos ou conhecidos previamente, o que conduz a uma diminuição do tempo de transmissão. Como se encontra demonstrado na Figura 2.15, neste modo o *header* é removido sendo que o tamanho do *payload*, *Error Coding Rate* e a utilização do CRC associado ao *payload* necessitam de ser configurados manualmente no recetor e transmissor [25].

2.5.8.4 PHY Payload

O formato de mensagens, tanto de *uplink* como *downlink*, possuem um PHY Payload, o qual a sua estrutura é inicializado por um MAC *header*, seguido pelo MAC *payload* e por ultimo um *Message Integraty Code* (MIC), utilizado na autenticação da mensagem, como se encontra representado na Figura 2.16.

PHYPayload		
1 Byte	1...M (59-230) Bytes	4 Bytes
MHDR	MACPayload	MIC

Figura 2.16: Estrutura do PHY *Payload* (Adaptado de [25]).

O MAC *header* tem como função especificar qual o tipo de mensagem (MType) enquanto o campo major é responsável por identificar o formato utilizado durante o processo de introdução de um nó na rede. O campo MType, permite distinguir entre seis tipos de mensagens, como se encontra representado na Figura 2.17, podendo estas ser divididas em duas classificações distintas: *data messages* e *join-request/accept*. A confirmação de mensagens de dados (*confirmed-data message*) é conseguida através do envio de um *acknowledge* por parte do recetor sendo verificado o oposto no tipo *unconfirmed-data message*. Como já foi referido anteriormente, um dos métodos de introdução de um nó na rede é através do *Over The Air Activation* no qual é necessário a realização de um procedimento introdutório de troca de mensagens, entre nó e o servidor de rede, o que conduz à utilização do tipo de mensagens *join-request* e *join-accept* [25].

MHDR		
Bit 7 ao 5	Bit 4 ao 2	Bit 1 ao 0
MType	RFU	Major

Figura 2.17: Estrutura do MAC *header* (Adaptado de [25]).

2.5.8.5 MAC Payload

O MAC *payload* contido na mensagem de dados, é constituído por um *frame header* e por dois campos opcionais: *Port field* (FPort) e *Frame Payload field* (FRMPayload). O campo FPort necessita de ser incluído quando contém um valor associado, se este for zero o FRMPayload irá conter MAC Commands, sendo valores entre 1 e 223 para aplicações especificas e os restantes, 244 até 255, reservados para uso futuro. O *Frame header* contém o endereço do dispositivo (DevAddress), um campo denominado *Frame Control* (FCtrl), seguido de um campo de 2 bytes *Frame Counter* (FCnt) e tendo por último a *Frame option* de 15 bytes utilizado para inserir os comandos MAC, caso sejam utilizados. As seguintes imagens representam as subdivisões do campo FCtrl para as situações de *downlink* e *uplink*.

MACPayload		
7...23 Bytes	0...1 Bytes	0...N (51-222) Bytes
FHDR	FPort	FRMPayload

Figura 2.18: Estrutura do MAC *Payload* (Adaptado de [25]).

As redes equipadas com tecnologia LoRa, permitem a utilização de qualquer *data rate* para os dispositivos incorporados na mesma, a introdução do LoRaWAN, como camada MAC, conduziu a uma expansão desta características tornando-a adaptável e otimizada consoante o nó. Deste modo, surge o *Adaptive Data Rate* (ADR), o qual faz parte do campo FCtrl, que quando este se encontra em utilização torna possível o controlo do *data rate*, nos nós, por parte da rede levando assim a uma otimização em termos de bateria do *nó* e capacidade da rede .

FHDR			
4 Bytes	1 Byte	2 Bytes	0...15 Bytes
DevAddr	FCtrl	FCnt	FOpts

Figura 2.19: Estrutura do *Frame header* (Adaptado de [25]).

Tal como já foi analisado anteriormente, é possível transmitir mensagens pela qual é necessária a confirmação de receção da mesma. Deste modo, é necessário ativar o campo ACK como se encontra representado na imagem 2.20, garantindo assim que em situações na qual o nó envia uma mensagem é enviado um *acknowledge* através de uma das janelas de receção do nó.

FCtrl				
Bit 7	Bit 6	Bit 5	Bit 4	Bits 3 ao 0
ADR	ADRACKReq	ACK	RFU	FOptLen

Figura 2.20: Estrutura do *Control Field* no envio de *uplinks* (Adaptado de [25]).

FCtrl				
Bit 7	Bit 6	Bit 5	Bit 4	Bits 3 ao 0
ADR	ADRACKReq	ACK	FPending	FOptLen

Figura 2.21: Estrutura do *Control Field* no envio de *downlinks* (Adaptado de [25]).

A *frame-option length* é utilizada com o intuito de informar acerca do tamanho do campo FOpts presente no *Frame Header*. O FPending é apenas utilizado em comunicações *downlink*, tendo como objetivo informar que existem dados pendentes e como tal existe a necessidade de abrir uma nova janela de receção de forma a transmitir esses mesmos dados [25].

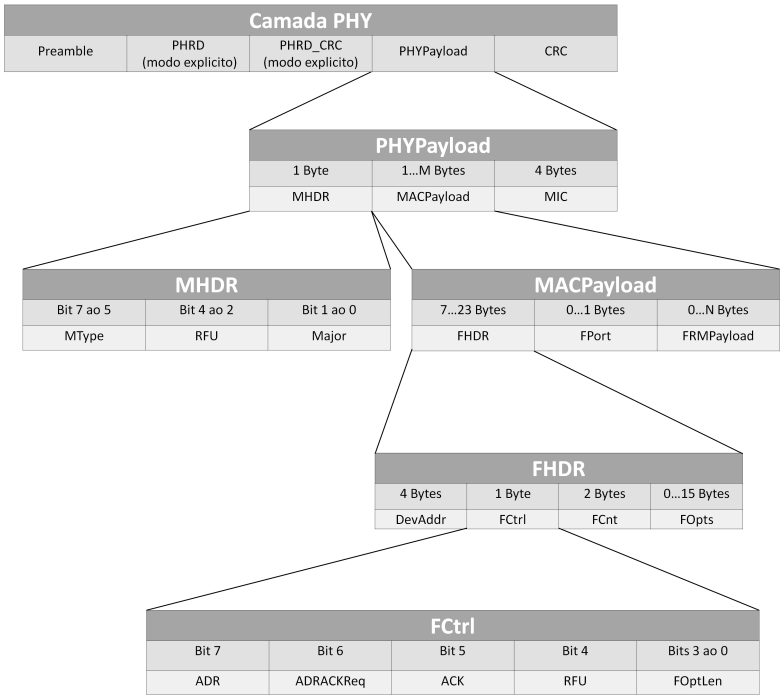


Figura 2.22: Estrutura final da camada PHY (Adaptado de [25]).

A seguinte tabela representa os diferentes valores selecionáveis de *Data Rate* tendo em conta o tamanho da mensagem. Assim, a terceira coluna representa a velocidade de envio, seguindo-se o tamanho possivel tendo em consideração as configurações presentes na segunda coluna.

Tabela 2.2: Tamanho máximo do MAC *Payload* consoante diferentes configurações.

Data Rate	**Configurações**	**Bits/s**	**Tamanho máximo *payload* (bytes)**
0	SF12/125kHz	250	59
1	SF11/125kHz	440	59
2	SF10/125kHz	980	59
3	SF9/125kHz	1760	123
4	SF8/125kHz	3125	230
5	SF7/125kHz	5470	230
6	SF7/250kHz	11000	230
7	FSK: 50kbps	50000	230

2.5.9 Aplicações LoRaWAN

Atualmente e tendo como base os dados apresentados no capítulo de motivação, o número de aplicações IoT tem vindo a crescer exponencialmente sendo cada vez mais a procura por soluções

que cumpram os requisitos impostos pelas mesmas. Deste modo, as tecnologias LPWAN permitem responder as exigências do mercado atual através das características *low power* e *long range* oferecendo diversos serviços consoante a tecnologia a optar. Sendo o LoRaWAN uma das tecnologias mais recorrentes, torna-se importante compreender qual o leque de aplicações destinado ao LoRaWAN tendo em conta as características desta mesma tecnologia. Assim, a presente secção tem como objetivo determinar em que aplicações o LoRaWAN pode ser enquadrado tendo como base toda a análise realizada nos capítulos anteriores [31].

2.5.9.1 Smart Metering

Uma das aplicações com uma grande tendência de utilização futura são as *smart metering* que atualmente é estimado a existência de 700 milhões de dispositivos destinadas a estas aplicações, sendo maior parte destes situados na China. Os *smart meter*, usualmente transmitem pacotes de dimensões reduzidas tendo como exemplo medições do consumo de água ou nível do gás bem como temperatura e humidade, onde os sensores transmitem esta informação apenas escassas vezes ao longo do dia e em intervalos de tempo elevados.

Este tipo de aplicações exige uma grande capacidade de rede, devido ao elevado número de dispositivos conectados, um baixo consumo energético por parte dos nós, um grande alcance e um baixo custo monetário no fabrico dos componentes (grande número de nós). Os requisitos mencionados coincidem com as características oferecidas pela tecnologia LoRaWAN, que apresenta apenas a desvantagem do preço elevado do gateway, porém este componente não possui necessidade de estar presente em grandes quantidades devidos as funcionalidades a que esta destinado como já foi abordado [32].

2.5.9.2 Tracking

Relativamente a aplicações destinadas ao *tracking* de objectcos, é necessário uma tecnologia capaz de fornecer uma boa performance em mobilidade auxiliada de um baixo consumo energético. Estas aplicações podem ser *outdoor* ou *in-storage*, como realizar o *tracking* de um objeto num armazém ou fornecer a localização de um objeto em constante movimento ao longo do transporte do mesmo ate ao destino. Deste modo torna-se relevante optar por uma tecnologia capaz de oferece um baixo consumo energético, um alcance elevado e uma mobilidade relativa. O LoRawan, através do posicionamento de gateways em diferentes locais e a capacidade de *end-devices* comunicarem com diversos gateways, sem a necessidade de estarem associados, possibilita a mobilidade requisitada por parte deste tipo de aplicações [33].

2.5.9.3 Agricultura

No âmbito da agricultura, a utilização de tecnologias LPWAN revela-se uma vantagem possibilitando não só a monitorização de campos agrícolas, mas também dos animais presentes nestes campos. Através de sensores adequados é possível determinar consumos de água, estado do solo, temperatura, localização de animais e parâmetros relacionados com a saúde do mesmo. A monitorização destes fatores permite obter uma maior eficiência, capaz de reduzir o impacto ambiental, minimizar custos como utilização de água através de rega automática tendo como base os valores obtidos pelos sensores. Algumas das aplicações desta área, como a mencionada, requerem o envio de *downlinks*, contudo estas são invulgares sendo as mensagens de *uplink* as mais usuais tendo como objetivo a monitorização de fatores como já referidos. Novamente o LoRaWAN enquadra-se como tecnologia indicada para estas aplicações [34].

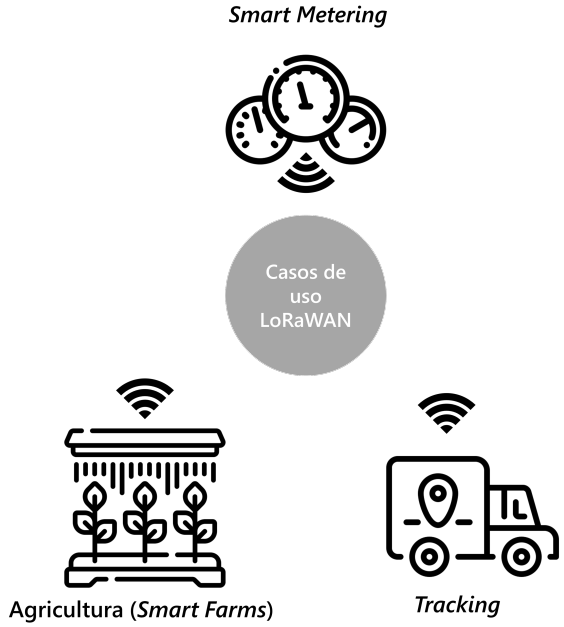


Figura 2.23: Representação gráfica dos três casos de usos mencionados.

2.5.10 Limitações LoRaWAN

Ao longo do presente documento, foram apresentadas em detalhe as capacidades da tecnologia LoRaWAN e como esta pode ser enquadrar em diversos meios, sendo assim uma das tecnologias LPWAN mais utilizadas atualmente. Porém, existem limitações associadas que se revelam importantes na compreensão total da tecnologia em questão. Deste modo, o presente subcapítulo destina-se a analisar quais as limitações e de que forma estas podem ser ultrapassadas ou minimizadas.

2.5.10.1 Duty Cycle

O *duty-cycle* de uma rede LoRa, é definido como a percentagem máxima de tempo que um dispositivo, neste caso nó, pode ocupar um determinado canal do gateway. Para o projeto desenvolvido, bem como para todas as redes presentes na Europa, este limite encontra-se com um *duty cylce* máximo de 1%.

$$Ts = Ta(\frac{1}{d} - 1) \tag{2.3}$$

Equation 2.3: Tempo de transmissão do nó.

Sendo d o valor de *duty cycle* máximo e que o tempo de transmissão de um pacote, *Time on Air*, é denotado como Ta é possível obter o período mínimo de tempo que um nó deve permanecer sem transmitir [35].

A partir da Figura 2.24 é possível analisar o *Time on Air*, utilizando um *code rate* de $4/5$ e uma largura de banda de 125 kHz para cada *spreading factor* variando o tamanho do pacote. É de notar, tal como já foi referido, que quanto maior o SF maior será o alcance obtido, contudo, o *time on air* torna-se assim mais elevado consequentemente aumentando o tempo em que o dispositivo esta sem transmitir. De acordo com o protocolo LoRaWAN, um nó com um determinado SF não pode exceder uma taxa de transmissão obtida através nd/Ta, sendo n o número de canais e Ta o *time on air* associado a um determinado *spreading factor*. Tendo como exemplo situações nas quais os dispositivos da rede possuem uma taxa de transferência máxima possível, é de notar que o número de pacotes recebidos, com sucesso, vai diminuindo à medida que são acrescentados dispositivos. Isto deve-se ao facto de existirem colisões na receção, ou seja, nós que transmitem no mesmo SF, canal e a altura de envio tendem a perder a sua informação, sendo que a probabilidade destes acontecimentos surgirem são cada vez mais elevados consoante o numero de nós existentes na rede [36].

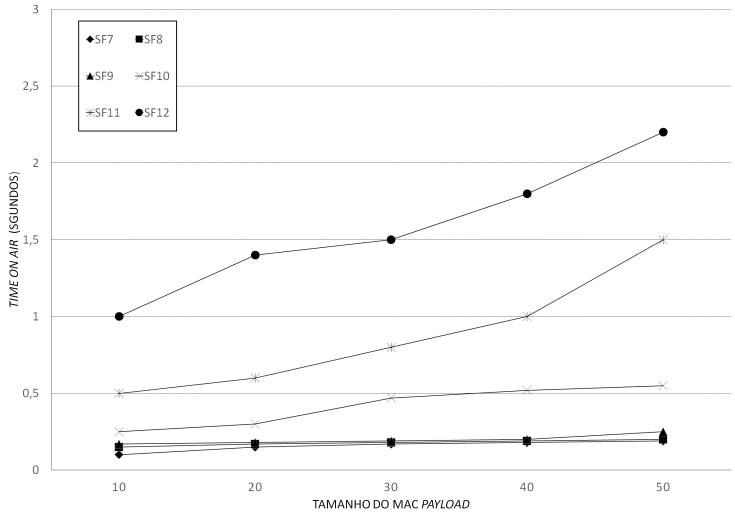


Figura 2.24: Gráfico baseado no estudo apresentado no artigo [35].

2.5.10.2 Fiabilidade

A fiabilidade de uma rede LoRa, é obtida através do uso de mensagens confirmadas e garantido que o gateway recebe na totalidade toda a informação enviada por parte dos nós. Para as diversas classes apresentadas a receção de *downlinks* é realizada de diferentes métodos sendo que para classes A apenas pode receber nas duas janelas após um *uplink*, classe B permite a sincronização de uma janela de receção enquanto que a classe C possibilita a receção de mensagens em qualquer momento. Desta forma, a fiabilidade encontra-se dependente do tipo de dispositivo da rede, pois pertence a esta propriedade a forma como é recebido um *acknowledge* por parte do gateway.

Um dos objetivos pretendidos pelo LoRaWAN passa por garantir o maior número possível de nós por gateway, ou seja, o gateway deve ser capaz de receber os dados enviados por todos os *nós*. Porém, o gateway é incapaz de estabelecer comunicação com os *nós* nos momentos em que se encontra a transmitir (como é o caso do envio de *acknowledge*) ou seja para situações onde o gateway realiza transmissões em 10% do seu funcionamento total, ao longo desse tempo encontra-se impossibilitado de receber informação dos nós.

Esta limitação põe em causa a fiabilidade da rede, conduzindo a perdas de informação que podem conter conteúdo fulcral da aplicação. A aplicação deve, assim, ter em consideração quais os requisitos associados à mesma de modo a garantir o máximo de fiabilidade possível. Medidas como a utilização reduzida de *acknowledges* ou mensagens *downlink* de modo a evitar transmissões por parte do gateway e um *data rate* eficiente com o objetivo de diminuir o *time on air*, representam um aumento na métrica em questão. Assim, estas limitações existentes condicionam o leque de aplicações que recorrem à tecnologia LoRaWAN, sendo necessário ter em consideração durante o processo de dimensionamento da rede de modo a garantir o máximo de fiabilidade possível sem qualquer perda de informação [37].

2.5.11 Vulnerabilidades LoRaWAN

Como foi abordado previamente, é possível verificar que, no que toca a segurança o LoRa apresenta soluções com a capacidade de evitar ataques como *man in the middle*, que colocam em risco a confidencialidade e integridade dos dados, entre outros. Esta tecnologia permite também adição de nós de forma segura através de dois métodos distintos: OTAA ou ABP. Porém, a segurança da rede não se encontra apenas dependente das implementações realizadas por parte do LoRaWAN, sendo o gestor da rede também responsável por assegurar a segurança da mesma. Não sendo o foco desta dissertação o estudo pormenorizado de possíveis vulnerabilidades, é porém necessario mencionar de forma breve algumas destas, realçando ataques físicos existentes. Deste modo, o presente capítulo tem como objetivo explorar vulnerabilidades de uma rede LoRaWAN de modo fornecer uma melhor compreensão da tecnologia.

2.5.11.1 Interferências

O crescimento da utilização de redes LPWAN apresenta um futuro promissor, tal como ja foi referenciado nos capítulos anteriores, e consequentemente ao surgimento de redes, independentes, com diferentes tecnologias LPWAN, incluindo LoRa. Este aumento, conduz à introdução de uma adversidade denominada de interferência causada pela implementação de diversas redes com proximidades elevadas.

Uma das aplicações das tecnologias LPWAN enquadra-se no âmbito das *smart cities*, onde é possível a existência de varias redes e como tal uma grande probabilidade de interferências sendo, assim, necessario uma gestão eficiente destas redes. Atualmente os módulos LoRa presentes no mercado, tem a capacidade de recorrer a transições ortogonais com a possibilidade de alterar as suas definições como a frequência, *spreading factor* ou *bandwidth* (parâmetros mencionados anteriormente) de modo a evitar interferências. Porém, este método não se revela viável em alguns aspetos pois a mudança de parâmetros como os mencionados anteriormente, causa impactos no alcance, consumo energético e na fiabilidade. A mudança destas variáveis de forma dinâmica nos sistemas LoRa atuais, é um processo complexo e requer em grande parte uma cooperação da rede.

2.5.11.2 Gestão de chaves

O posicionamento de nós da rede em locais estratégicos, é um fator fundamental consoante o propósito da aplicação a qual estão destinados, porém a escolha do local do dispositivo é também influenciado por possíveis ataques físicos. Considerando que a ameaça parte pelo acesso ao hardware do nó, torna-se possível obter as chaves mencionadas nos capítulos anteriores (AppKey, AppSKey e NwkSKey) através de ataques como *side channel analysis*. Este tipo de ataques, recorrem a variações energéticas ou emissões eletromagnéticas (*channels*) produzidas por parte

do dispositivo e que se encontram relacionadas com operações que este executa ao longo do seu funcionamento. Deste modo, quando um nó recebe dados encriptados e é executada a desencriptação do mesmo, o sistema produz *side channels* que são intercetados e juntamente com os dados recebidos, pelo nó, torna possível obter as chaves utilizadas na encriptação dos *payloads*. Assim, caso o atacante possua o acesso a estas chaves únicas, os pacotes podem ser intercetados ou torna-se até mesmo possível o envio de informação falsificada.

Os ataques físicos, passam também pela possibilidade de modificação do hardware presente no nó quando este está no alcance do atacante. De modo a transmitir informação, os dispositivos encontram-se equipados com *transceiver*, RN2483 por exemplo, que por sua vez estão conectados a um microcontrolador, através de UART, com o objetivo de encriptar e transmitir a informação. As chaves encontram-se armazenadas nestes *transivers modules* e como tal são abstraídas do micontrolador, sendo este apenas responsável por enviar os dados para o modulo, que são posteriormente encriptados e transmitidos. Deste modo, uma das forma de ataque passa pela substituição do MCU ou através do pinos UART, enviar informação criada pelo atacante, com o intuito de falsificar os dados conduzindo à introdução de valores com a possibilidade de alertar o sistema para uma situação de falso risco (analisando o caso dos incêndios, o sistema poderia iniciar as medidas de alertar) [38].

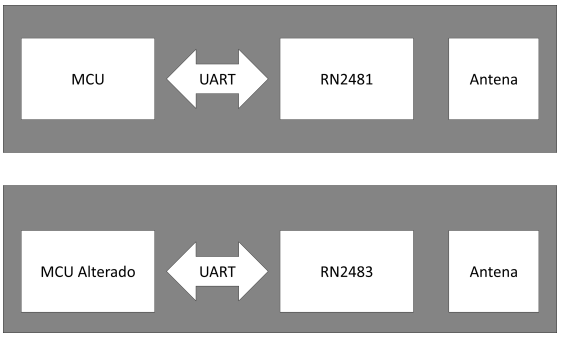


Figura 2.25: Exemplo de um possível ataque físico a um nó. Este tipo de ataques requer o posicionamento dos nós em locais estratégicos e equipados com um encapsulamento seguro.

2.5.11.3 Servidor de rede

Como tem vindo a ser referido ao longo do presente documento e com maior relevo nos subcapítulos anteriores, as chaves utilizadas para garantir a encriptação e integridade dos dados são armazenadas no servidor da respetiva rede. Este fator introduz uma nova vulnerabilidade que ultrapassa a segurança oferecida pela tecnologia LoRaWAN, conduzindo a um acesso indevido ao servidor no qual estão contidas as chaves. É assim necessário restringir o acesso de terceiros que não possuam privilégios, impossibilitando a leitura e escrita das chaves mantendo assim o seu propósito: desencriptação e verificação. A disponibilidade constante do servidor de rede é fulcral em todo o funcionamento da mesma tendo este de possuir uma resiliência elevada

a ataques como *Denial of Service* (DoS). Uma falha deste tipo no servidor, conduz à incapacidade da receção de informação transmitida pelos gateways e consequentemente a comunicação para com os nós da rede. A origem deste ataque esta geralmente associado as interface web responsáveis por efetuar comunicações com o servidor.

2.6 Gateways

Após uma breve análise relativa à arquitetura de uma rede LoRaWAN realizada na secção anterior, é identificar quais os três componentes fulcrais e distintos para o total funcionamento de uma rede LoRaWAN: nó, gateway e servidor de rede. Os três componentes mencionados serão abordados em maior detalhe posteriormente, porém e sendo o foco desta dissertação a criação de um *custom* gateway, torna-se necessário realizar um levantamento dos gateways presentes no mercado de modo a conseguir obter um dos motivos que conduzem à construção de um novo gateway. O subcapítulo em questão tem assim o objetivo de fornecer um estudo relativo aos produtos mencionados, realçando as suas características principais e de que forma estas influenciaram o gateway final obtido.

2.6.1 LORIX One

O Lorix One é um dos primeiros gateways *low cost* presentes no mercado indicados para ambientes *outdoors*. Apresenta características como dimensões reduzidas, performance elevada, com facilidade no seu posicionamento e adaptável a diversos climas e ambientes (IP65 *waterproof*, suporte de temperaturas entre -30 e +55 graus). Este gateway, permite ao utilizador criar uma rede LoRa económica com um coordenador incorporado de um sistema Linux de fácil interação com a *cloud* Loriot. Possui também suporte para cartões SD e a capacidade de suportar interface Ethernet [39].

2.6.2 Conduit

MultiConnect Conduit é outro gateway com características distintas, apresentando-se como versátil, escalável e totalmente configurável. Toda a configuração deste dispositivo pode ser executada remotamente através da *cloud DeviceHQ*, obtendo assim melhor performance. Em termos de conectividade, este possui como standard Bluetooth e Wi-Fi, porém tem a capacidade da introdução de mCards que possibilitam a utilização de outro meios de comuniação sem e com fios. Um destes mCards disponíveis, é o LoRaWN mCard capaz de alterar este coordenador de modo a suportar comunicações de longo alcance com mDot e xDot equipados com sensores. Além destas características, possui também um sistema Linux, *open source*, de fácil configuração

e gestão, tornado o Conduit um gateway idealizado para aplicações IoT com os mais diversos requisitos [40].

2.6.3 Wirnet

O Wirnet iBTS distingue-se dos restantes gateways devido ao facto de possuir uma grande versatilidade. A capacidade de conseguir ser facilmente modificado a nível de hardware, através de módulos desenvolvidos pela empresa, auxiliado de um processamento poderoso torna esse gateway um dos melhores do mercado atribuindo características como escabilidade e adaptação a qualquer projeto no qual são inseridos. Possui também a possibilidade de fornecer geolocalisação bem como dois modelos no que toca ao seu encapsulamento (metal e plástico) [41].

2.6.4 Lorrier R2

O Lorrier R2 é um gateway totalmente dedicado para espaço exteriores, baseado no protocolo LoRaWAN, garantindo alcances elevados seja a utilização do mesmo por operadores de telecomunicação ou redes locais, possuindo um revestimento em metal que torna o LR2 resistente a diversos ambientes (proteção IP66). Semelhante ao gateway desenvolvido ao longo da dissertação, o Lorrier recorre a um modulo de radio externo (iC88a) no qual se encontra incorporado o processador SX1301 possuindo um microcontrolador *BegleBone Green* (1GHz) conectado através de SPI ao modulo iC88a. A comunicação com o servidor de rede é estabelecida através do protocolo TCP/IP auxiliada de uma board *Mikrokit Routerboard*, capaz de fornecer configurações pormenorizadas relativas à comunicação entre o gateway e o servidor de rede. [42]

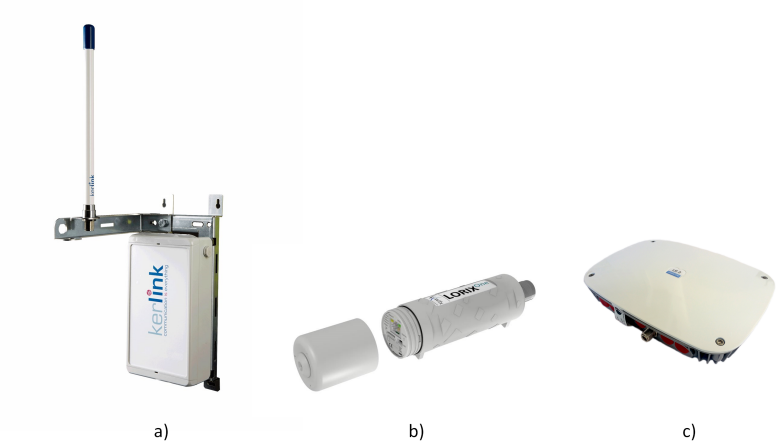


Figura 2.26: Gateways mencionados a) Wirnet; b) Lorrier R2; c) Conduit; [39] [41] [42]

2.6.5 Comparação dos Gateways

Após uma breve análise dos quatro produtos mais utilizados no mercado, quais as caraterísticas principais e diferenciando cada um destes, encontram-se reunidas as condições para elaborar uma conclusão relativamente a estes produtos. Um dos pontos fulcrais é o meio utilizado na comunicação com o servidor de rede, a partir de uma análise da tabela 2.3 é possível verificar que a tecnologia mais utilizada é Ethernet, seguindo-se o Wi-Fi e tecnologia móvel (4G, 3G e 2G). Alguns dos produtos apresentados, não possuem na totalidade estes meios como base, mas sim como sendo opcionais levando consequentemente a um valor acrescido do produto. O armazenamento representa um papel de menor importância no design do gateway, porém torna-se relevante uma avaliação acerca desta característica para aplicações onde existe a necessidade de proceder ao armazenamento de dados recebidos pelos nós. É de notar que existem dois métodos mais recorrentes: armazenamento externo e interno. A utilização de cartões SD e Micro SD torna o gateway mais flexível permitindo a mudança do armazenamento quando necessário em contrapartida as memorias flash, eMMc. Por último, a escolha do Sistema Operativo (OS) apresenta um menor relevo sendo possível verificar, de acordo com os gateways apresentados neste subcapítulo, uma utilização do Linux ou versão deste na totalidade dos produtos. Assim, a partir desta breve análise é possível retirar algumas características a ser incorporadas no gateway final sendo atribuído maior relevo aos meios de comunicação utilizados para com o servidor de rede e ao encapsulamento do componente.

Tabela 2.3: Comparação entre os diferentes gateways analisados.

	Lorix	**Conduit**	**Wirenet**	**Lorrier**
Comunicação	Ethernet	Ethernet, Wi-Fi, 4G,3G	Ethernet, Wi-Fi, 2G,3G	Ethernet, Wi-Fi (opcional), 4G,3G (opcional)
Armazenamento	Cartão SD	Cartão Micro SD	8 GB eMMc	4 GB eMMc
Sistema Operativo	Linux	mLinux	Linux	Linux

2.7 Conclusão

O presente capítulo abordou, para além do estado da arte relativo as tecnologias LPWAN e gateways LoRaWAN, alguns dos conceitos que serão necessários nos restantes capítulos. Foi também analisado a tecnologia utilizada no desenvolvimento de todo o projeto, apresentando as diferentes camadas e algumas vantagens e limitações do LoRa. A comparação entre tecnologias e gateways, permite assim compreender as opções realizadas no trabalho efetuado durante a dissertação. Assim, a necessidade de realizar modificações no software e a de adicionar periférico conduz à construção de um gateway desde raiz tendo em consideração todo os estudo realizado no presente capítulo.

Capítulo 3

Especificações do Sistema

Ao longo dos capítulos anteriores foram abordados conceitos relativos a redes LPWAN, com foco no LoRaWAN pelo facto de esta ser a tecnologia utilizada na elaboração do projeto. Uma rede LoRa é composta por três componentes principais (nó, gateway e servidor de rede), como tem vindo a ser referenciado ao longo do documento, porém para a dissertação o componente principal é o gateway. Assim, antes de documentar de que forma foi obtido o protótipo final, torna-se necessário referenciar quais os principais componentes utilizados, a constituição da arquitetura do sistema e uma abordagem geral das duas *boards* : Microchip Radio *Board* e *Custom Board*. Este capítulo pretende assim, responder aos tópicos mencionados através de uma divisão em três subsecções: Requisitos, Arquitetura e Componentes do Sistema

3.1 Requisitos do Sistema

Antes de iniciar a fase de desenvolvimento, é necessario ter em consideração os requisitos que o sistema deve cumprir. Estes baseiam-se em atributos, já mencionados ao longos do documento, referentes aos gateways de uma rede LoRaWAN : fiabilidade, flexibilidade e redundância. Os pontos seguintes, abordam os requisitos impostos e como são obtidos os 3 atributos mencionados:

- **Redundância nas comunicações:** o sistema deve possuir a capacidade de comunicar com o servidor de rede através de uma ou mais tecnologias de comunicação (Wi-Fi, Tecnologia Móvel ou Ethernet) com o intuito de conferir maior fiabilidade ao sistema.
- **Armazenamento de dados:** o sistema deve possuir a capacidade de armazenar dados para situações onde se verifique a inexistência de conexão com o servidor de rede, de modo a preservar estes dados sendo posteriormente transmitidos.
- **Encapsulamento:** o sistema deve ser protegido de ataques físicos e possíveis danos causado por acontecimentos naturais.
- **Reencaminhamento de informação:** o sistema deve possuir a capacidade de transmitir informação recebida por parte dos nós, através dos 8 canais possíveis, bem como receber e enviar os dados transmitidos pelo servidor de rede.

3.2 Arquitetura do Sistema

O sistema referente ao gateway encontra-se dividido em duas *boards* com a capacidade de comunicarem entre si tendo como objetivo a transferências de pacotes LoRa. Como já referido anteriormente, o gateway tem como funcionalidade principal o reencaminhamento de pacotes, recebidos pelos nós, para o servidor de rede ou vice-versa. O produto final obtido, deve assim ter esta função básica sendo obtida pela junção das duas *board* já referidas: *Custom Board* e *Microchip Radio Board*.

A *Custom Board*, é responsável por estabelecer a comunicação com o servidor de rede, enviando ou recebendo pacotes do mesmo, e pela transferência de dados para com a Microchip *Radio Board*, desenvolvida pela Microchip. A Figura 3.1, representa a arquitetura final do sistema que inclui a *Custom Board* e a *Microchip Radio Board*, os dois seguintes subcapítulos abordam em maior detalhe o funcionamento destas *boards* sendo posteriormente analisado com mais pormenor os componentes fulcrais que fazem parte das mesmas.

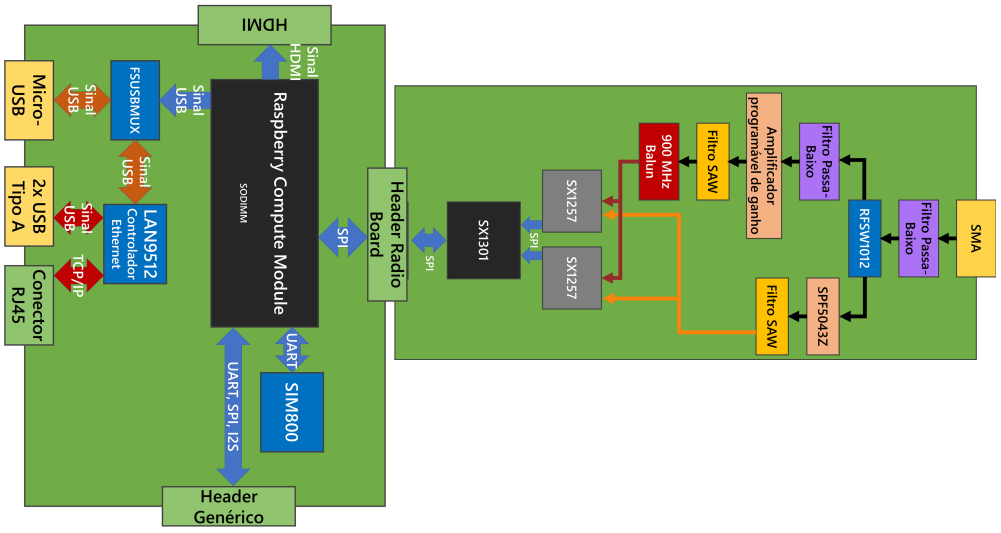


Figura 3.1: Diagrama de blocos do sistema final. Representa as ligações entre os diversos periféricos e as *boards* em análise .

3.2.1 *Custom Board*

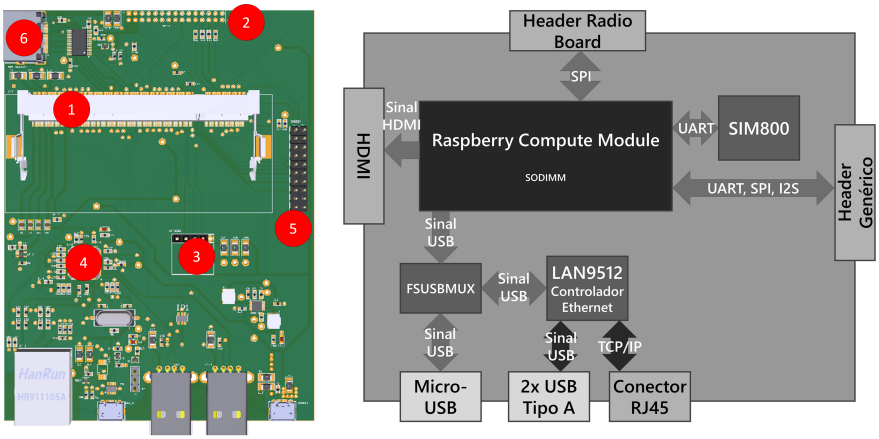


Figura 3.2: A *Costum Board* desenvolvida ao longo da dissertação e o diagrama de blocos respetivo.

1: Compute Module SODDIM 2: SPI Header 3: SIM800
4: LAN9512 5: General Header 6: HDMI

A *Custom Board* representa a board desenvolvida para o projeto em questão, tendo como função principal reencaminhar os pacotes, recebidos por parte da *Radio Board*, para um servidor de rede. A Figura 3.2 representa o diagrama de blocos e a *board* respetiva com os módulos de maior relevo assinalados.

O sistema centra-se no Raspberry Compute Module 3 (CM3) o qual está incorporado com um sistema operativo Raspbian. O CM3 estabelece a comunicação com os diferentes periféricos, como está representados na Figura 3.2, recorrendo em algum dos casos a protocolos como SPI ou UART sendo também responsável pela gestão dos pacotes e comunicações para com o servidor de rede. Este componente é inserido na board através de um conector SODDIM (elemento 1 da Figura 3.2).

A comunicação para com a *Radio Board* é realizada a partir do CM3 através do protocolo SPI. Fisicamente, a ligação entre ambas as *boards* é estabelecida utilizando o *header* presente na Figura 3.2 e referida pelo número 2.

A *Custom Board* encontra-se também equipada com duas portas USB tipo A, duas micro-USB, para conexão de periféricos, e um conector RJ45 utilizado no suporte à Ethernet. O controlo e interface destes é realizado recorrendo aos integrados LAN9512 (elemento 4) e FSUBMUX, que serão abordados em maior detalhe no seguinte capítulo, e por sua vezes conectados ao CM3.

O elemento 5 é um *header* de vinte pinos que permite realizar a comunicação com periféricos externos que utilizem protocolos SPI, UART ou I2C estando conectado diretamente ao CM3. O *header* restante, elemento 3, recorre ao protocolo UART e tem com finalidade integrar o SIM800 na *Custom Board*.

Por último, o elemento 6 remete para o conector HDMI também ele diretamente conectado ao CM3. Esta ligação é realizada através dos sinais HDMI presentes no MCU. A seguinte tabela sumariza as funções de cada um dos elementos mencionados.

Tabela 3.1: Função dos módulos e *headers*.

Módulos e Headers	Função
LAN9512	Em conjunto com as portas USB e o conector RJ45, permite a ligação de periféricos externos e a um servidor de rede recorrendo ao Ethernet
SIM800	Permite a ligação ao servidor de rede e o envio de mensagens através da rede móvel
CM3 SODDIM	Conector SODDIM onde é inserido o CM3. Esta é a unidade de processamento da *Custom Board*
HDMI	O módulo HDMI fornece o acesso à interface gráfica com o intuito de auxiliar nas configurações da board
General Header	Este header permite a ligação de periféricos que utilizem protocolos de comunicação como UART, I2C ou SPI
Radio Board Header	A conexão com a *Radio Board* é estabelecida através deste *header* que recorre ao protocolo SPI

3.2.1.1 Raspberry Pi Compute Module

O Raspberry Pi Compute Module 3 possui um processador BCM2837, com uma velocidade de 1.2GHz (quad-core), uma RAM de 1Gb e uma memória flash de 4Gb. Outra característica de relevo são as dimensões reduzidas de 67.6mm x 31mm e o modo como é realizado a inserção através de um conector DDR2 SODIM (igual aos utilizados nas memórias RAM) na *board* desenvolvida. Outra vantagem comparativamente à Raspberry Pi, provem da possibilidade de utilização de um maior número de GPIOs e interfaces, tal como a capacidade de ser utilizada em diversas boards, com o suporte adequado, garantindo assim uma grande flexibilidade [43].

Tabela 3.2: Diversos periféricos possíveis de utilizar no *compute module*.

48x GPIO	2x SD/SDIO	1x NAND interface
2x I2C	1x HDMI	1x 4-lane CSI Camera Interface
2x SPI	1x USB2 HOST/OTG	1x 2-lane CSI Camera Interface
2x UART	1x DPI	1x 4-lane DSI Display Interface

Figura 3.3: Raspberry Pi Compute Module 3 [43].

O CM3 possui 204 pinos com diferentes funcionalidades sendo necessário determinar quais a ligações corretas do CM3 com os restantes periféricos. Assim, seguinte tabela representa as conexões mencionadas, salientando os módulos mencionados na Figura 3.2.

Tabela 3.3: Conexões entre o compute module e os diversos periféricos presentes no prototipo final.

	Função	***Pins***
HDMI	SDA e SCL	173 e 175
	Data0- e Data0+	117 e 119
	Data1- e Data1+	123 e 125
	Data2- e Data2+	129 e 131
	Clock- e Clock+	111 e 113
	HPD	88
	CEC	17
LAN	NRST	21
Header Genérico	SDA e SCL	3 e 5
	SPI MOSI e MISO	65 e 69
	SPI CLK e CE	63 e 71
	UART Tx e Rx	51 e 53
Header Radio Board	SPI MOSI e MISO	29 e 33
	SPI CLK e CE	27 e 35
	RST	83
SIM800L	UART Tx e Rx	46 e 48
	RST	34

3.2.1.2 LAN 9512

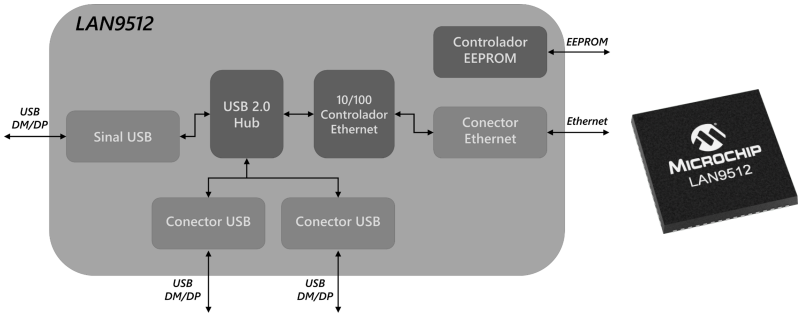


Figura 3.4: Integrado LAN9512 utilizado para realizar a interface entre o Compute Module 3 e a porta Ethernet tal como as portas USB (Adaptado de [44]).

O LAN9512 é um hub USB 2.0 de alta performance auxiliado de um controlador Ethernet 10/100, sendo um componente de *high performance* e baixo custo. A Figura apresentada, representa um diagrama de blocos do LAN9512 onde é possível verificar o conteúdo deste: um hub USB 2.0, dois portos *downstream* USB 2.0, um porto USB 2.0 *upstream*, um porto Ethenet, um controlador Ethernet e EEPROM.

O hub USB presente tem capacidade de suportar velocidades desde *Low-Speed*, *Full-Speed* e *High-Speed* dos dispositivos *downstrem*, conectados aos portos associados, possuindo também um *Transaction Translator* (realiza conversão dos tipos de USB) dedicado a cada porta USB, obtendo desta forma maior flexibilidade.

Para a *board* em questão, o controlador EEPROM não foi utilizado, este tem como funcionalidade conectar-se a uma EEPROM externa capaz de armazenar configurações, associadas ao USB e Ethernet, que são carregadas no momento em que a *board* é ligada ou sofre um *reset*.

O LAN9512, possui três modos distintos de operação sendo destinados ao controlo das portas USB e consequentemente determinam o consumo energético. Para o projeto desenvolvido, foi mantido o modo que vem por defeito (de menor consumo), ou seja, de acordo com o *datasheet*, o SUSPEND2 a partir do qual os pinos relativos ao sinal USB são apenas ligados quando necessário [44].

3.2.1.3 ECN28j60

O ECN28J60 foi utilizado como alternativa ao LAN9512, sendo este também um controlador Ethernet que por sua vez recorre a uma interface SPI, através do General Header. [45]

Figura 3.5: Modulo ENC28J60, exemplo que contém todos os componentes necessários para o funcionamento deste [45].

3.2.1.4 SIM800L

O SIM800L é um modulo quad-band GSM/GPRS capaz de funcionar em diferentes frequências: GSM850MHz, EGSM900MHz, DCS1800MHz e PCS1900MHz. Devido as dimensões reduzidas apresentadas por este modulo é possível que este seja facilmente incorporado na maior parte das aplicações destinadas a comunicações, como é o caso do projeto em questão. O SIM800L possui a capacidade de realizar chamadas, envio de mensagens e estabelecer comunicação com servidores ou outras aplicações que recorrem à Internet, porém para a dissertação em causa apenas as ultimas duas propriedades foram utilizadas. O CM3 estabelece comunicação com o SIM800 através do protocolo UART [46].

Figura 3.6: Representação do modulo SIM800 e restantes componentes que permitem este ser incorporado diretamente no gateway [46].

3.2.2 *Radio Board*

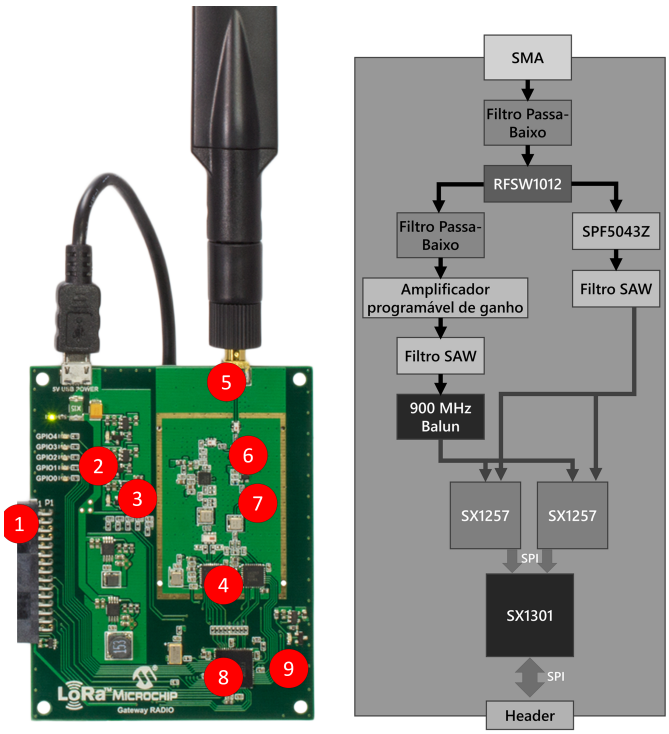


Figura 3.7: A *Radio Board* desenvolvida pela Microchip e o diagrama de blocos respetivo [47].

1: SPI Header	2: MCP1824	3: MCP1726
4: SX1257	5: Conector SMA	6: RFSW1012
7: SPF5043Z	7: SX1301	9: MCP1824

A segunda secção do sistema, é destinado à *board* de rádio desenvolvida pela Microchip com o intuito de transmitir ou receber informação utilizando a tecnologia LoRa. A receção ou emissão dos pacotes de informação, é realizado recorrendo a dois *transceivers* SX1257 sendo posteriormente armazenados no base *band processor* SX1301. Os dados são mais tarde transmitidos para a *Custom Board* através do SPI header (elemento 1) recorrendo ao protocolo SPI.

Todo o processo desde a receção dos pacotes até ao armazenamento dos mesmo, encontra-se ilustrado no diagrama de blocos da Figura 3.7. Inicialmente, o sinal recebido é filtrado e posteriormente demodulado pelos dois módulos SX1257 (elemento 4), sendo depois armazenado por parte do módulo SX1301 (elemento 7).

3.2.3 SX1257 e SX1301

O SX1257 é um front-end RF (*Radio Frequencie*) para digital equipado com a técnica I/Q de modulação/demodulação com a capacidade de transmitir ou receber informação e suportar técnicas de modulação como LoRa [48].

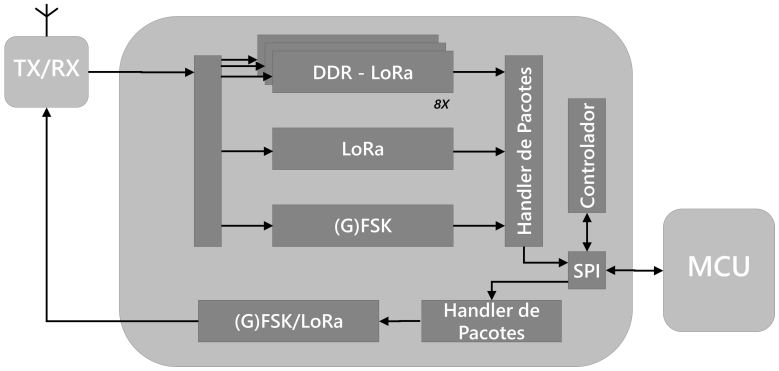


Figura 3.8: Diagrama de blocos do modulo SX1301 (Adaptado de [49]).

O integrado SX1301 é um *digital baseband chip* responsável por receber o *bitsream* I/Q de um ou dois *transceivers* (SX1257). Posteriormente, os pacotes são armazenados numa FIFO encontrando-se prontos para serem transmitidos quando o MCU realizar esse mesmo pedido. O controlo do integrado SX1301 é realizado com o auxílio de uma *Hardware Abstraction Layer* (HAL), desenvolvida pela Semtech, a partir do MCU [49] .

3.2.4 Seleção do Encapsulamento

Após o desenvolvimento do gateway a nível de hardware e software, este encontra-se preparado para ser posicionado em qualquer local, porém antes da execução deste processo é necessario ter em consideração o encapsulamento adequado do gateway. Sendo maioritariamente a utilização deste em locais no exterior, existem fatores a ter em consideração sendo o clima aquele que apresenta maior relevo.

Para o encapsulamento em questão, foi optado por um com IP 67 de modo a fornecer ao gateway hermética contra poeiras e proteção contra imersão temporária de água.

3.3 Aplicação Web

Antes de realizar a implementação da aplicação web, é necessário elaborar um diagrama de relações referente à base de dados recorrendo ao MySQL. A partir deste torna-se possivel obter uma melhor compreensão dos procedimentos a realizar para obter a aplicação final como é descrito no capítulo seguinte. Como tem vindo a ser mencionado ao longo do documento, as redes possuem nós e gateways, sendo para esta situação considerado que os nós possuem diversos sensores e o gateway encontra-se equipado com os periféricos mencionados nesta dissertação. Deste modo, os nós são representados com uma chave primaria, referente ao ID do mesmo, uma chave secundaria, associado ao ID da rede, e os restantes campos relacionados com os dados e a localização do dispositivo (latitude e longitude). O gateway possui uma estrutura semelhante à exceção do campo de dados, ou seja, uma chave primaria associado ao ID do gateway, uma

chave secundaria relativa à rede e as coordenadas do mesmo: latitude e longitude. A rede por sua vez possui, como ja mencionado, uma chave primaria referente ao ID da mesma, uma chave secundaria associada ao utilizador e o numero de redes estabelecidas. Por ultimo, o utilizador tem apenas como campo uma chave primaria referente ao ID do mesmo. A seguinte imagem representa a relação entre cada uma destas estruturas na base de dados.

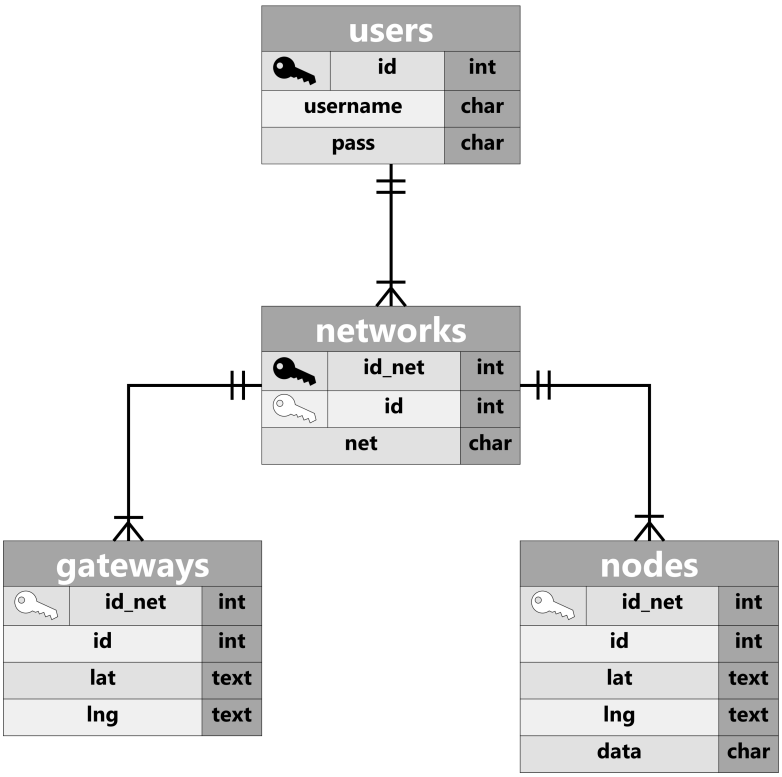


Figura 3.9: Diagrama de Entidades e Relações.

3.4 The Things Network

O componente principal pela gestão da rede na tecnologia LoRaWAN é o servidor da rede, onde é realizado o *fetch* de todos os pacotes recebidos e consequentemente o armazenamento destes. Porém o simples envio da informação do gateway para o servidor não é suficiente. Este último necessita de possuir a capacidade de interpretar os pacotes LoRaWAN, tendo em consideração a segurança relacionada com a tecnologia, e possibilitar a existência de aplicações associadas ao servidor para interpretar a informação, como é o caso da aplicação web desenvolvida nesta dissertação.

Deste modo foi utilizada a plataforma *The Things Network* (TTN), responsável por realizar o *routing* dos dados recebidos por parte do gateway, entre os nós e as aplicações. A TTN define as aplicações como sendo algo que o *nó* comunica na Internet, onde na dissertação em questão é destinada ao controlo de incêndios florestais. Esta plataforma oferece também um consola (TTN *Console*) que possibilita a adição dos dispositivos da rede, nós e gateways, permitindo alterar o método de inserção (OTAA ou ABP) e gerar as chaves necessárias [50] [51].

Para além das funcionalidades mencionadas, a TTN fornece também *Platform Integrations* com a capacidade de sincronizar os registos e dados provenientes por parte dos nós com plataformas como Azure IoT Hub, AWS IoT e IBM Watson. Uma das diversas *integrations* disponíveis e mais comum, é o reencaminhamento de dados para um *endpoint* (*HTTP Integration*), sendo o utilizado no projeto em causa [52].

3.5 Conclusão

Neste capítulo foram definidas as especificações de todo o sistema e realizada uma análise da tecnologia utilizada no projeto desenvolvido, LoRaWAN. No gateway foi utilizada a *Radio Board* fornecida pela Microchip auxiliada da *Custom Board*, sendo esta última o foco da dissertação. A *board* desenvolvida é equipada com uma CM3 e os restantes periféricos mencionados - HDMI, Ethernet, USB e GSM. Foi também designada qual a plataforma responsável por receber a informação transmitida pelo gateway, sendo para este fim utilizada a *The Things Network*. Por último, uma outra fase desta dissertação passa pelo desenvolvimento de uma aplicação web capaz de interpretar os dados recebidos e como tal, neste capítulo foi especificada a estrutura da base de dados utilizada e os requisitos da aplicação - a Figura seguinte representa a arquitetura final do projeto, diferenciando cada uma das camadas desenvolvidas.

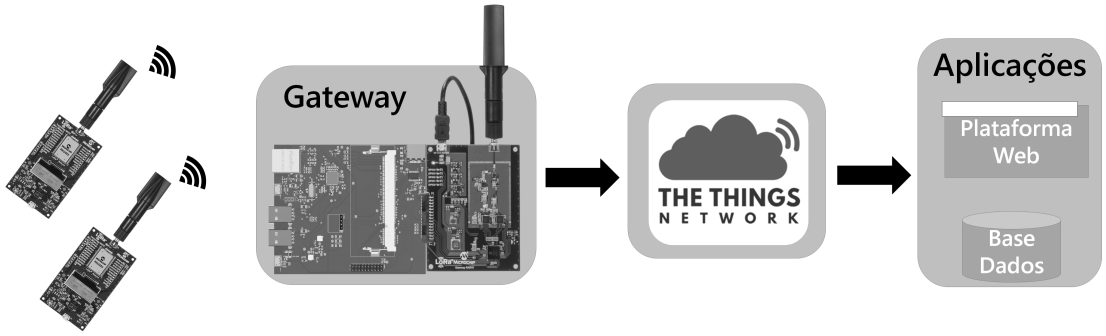


Figura 3.10: Representação e enquadramento de todos os componente presentes na rede, atribuindo maior relevo as secções desenvolvidas: gateway e aplicações.

Capítulo 4

Implementação

Após uma análise acerca das especificações do sistema, encontram-se reunidas as condições necessárias para realizar uma abordagem completa de todo o processo de implementação elaborado ao longo da dissertação em questão. O capítulo encontra-se assim dividido em dois subcapítulos: hardware e software. Relativamente ao hardware, são abordados os diferentes procedimentos desenvolvidos, design do esquemático e PCB, na obtenção da *Custom Board* realçando aqueles que apresentam maior relevo. Por último, o subcapítulo de software, apresenta de que forma é realizada a receção e reencaminhamento dos pacotes, para o servidor de rede, como é obtida a redundância nas comunicações e finalizando o subcapítulo, é abordada a aplicação web desenvolvida que visa realizar a interpretação da informação recebida.

4.1 Hardware

A implementação de hardware, é inteiramente dedicado ao processo de elaboração da *Custom Board* sendo esta uma das duas etapas fulcrais da dissertação. Ao longo do subcapítulo, serão apresentados os diferentes módulos no qual foi dividido todo o esquemático da *Custom Board* de cada um justificando de que modo foi obtido o protótipo final tendo como base os conceitos abordados no capítulo anterior. O final do subcapítulo, tem como objetivo apresentar uma conclusão acerca do tema em questão e o resultado final obtido.

Todo o projeto foi desenvolvido recorrendo à ferramenta *Altium* (esquemático e *layout* de PCBs) e com base no esquemático referente ao *IO Board* apresentada por parte da *Rapsberry Pi*. Cada módulo encontra-se interligado através do módulo CM3, onde neste estão presentes as ligações dos restantes ao pinos do CM3. Os blocos HDMI, USB e LAN abordam os respetivos conectores associados ao nome sendo o POWER dedicado apenas à descrição da geração de tensões presentes na *Custom Board*. Assim os seguintes pontos, realizam um análise detalhada relativo a cada um destes blocos e demonstrando de que forma estes se encontram ligados entre si.

4.1.1 Compute Module

O esquemático base e a partir do qual derivam os restantes ficheiros, encontra-se relacionado com o CM3. Possui a divisão do Compute Module em grupos distintos, o que a compreensão do esquemático geral, sendo através da figura 4.7 os conjuntos diferentes:

- Power – este bloco é destinado aos reguladores de tensão presentes na *Custom Board*.
- HDMI – bloco responsável pelo interface entre o conector associado e o CM3.
- *Header* Geral e SIM800 - referente aos headers SIM800 e de uso geral.
- USB e LAN – aborda os conectores USB e Ethernet, tal como o integrado associado (LAN95120).

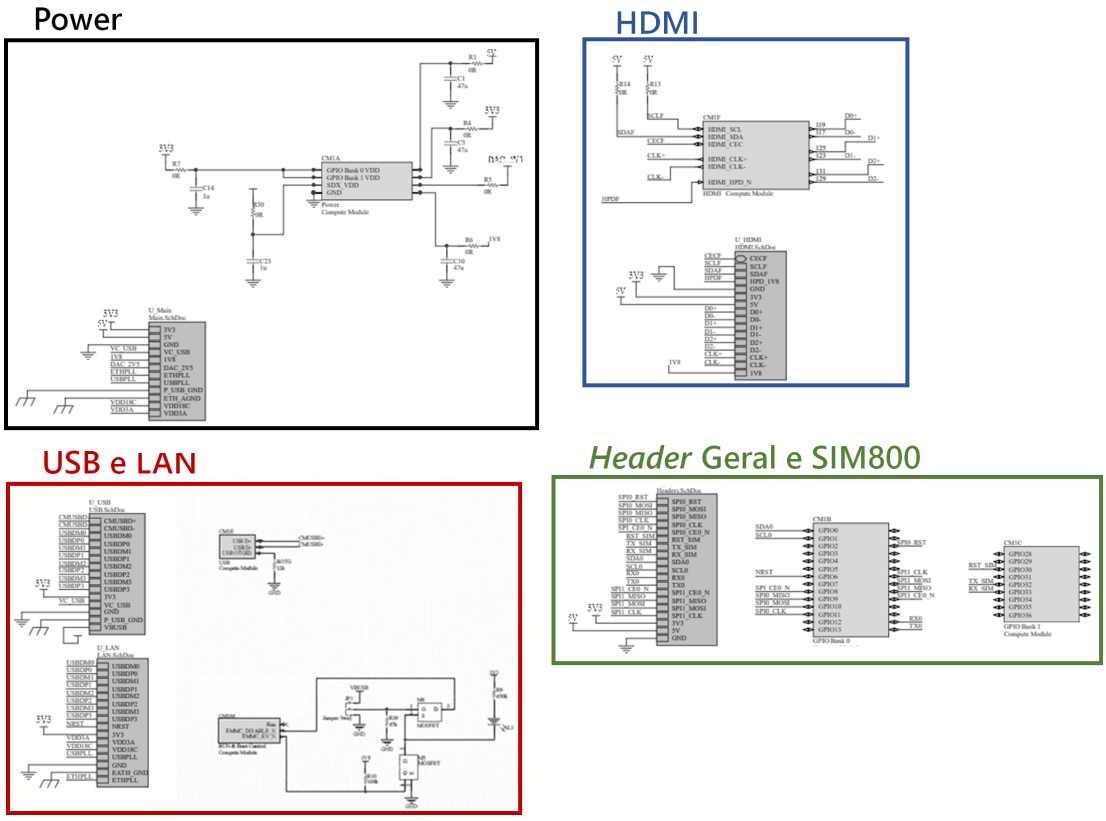


Figura 4.1: Representação geral dos módulos desenvolvidos e como estes estão interligados.

4.1.2 Power

Como já foi mencionado anteriormente, a *board* desenvolvida necessita de quatro níveis de tensão distintos sendo três delas utilizadas em grande parte na alimentação dos integrados presentes na placa: 3.3, 1.8, 2.5 e 5 Volt. Assim, com o intuito de obter estes valores de tensão foram utilizados dois reguladores de tensão sendo para os dois primeiros valores mencionados, 3.3 e

1.8, o PAM2306 e para 2.5 Volt o AP7115, sendo os 5 Volt obtidos a partir da tensão fornecida pela porta micro USB de alimentação.

Analisando o regulador PAM2306, este é capaz de suportar uma tensão de entrada no intervalo de 2.5 até 5.5 Volt, sendo para a situação em estudo utilizado 5 Volt provenientes da porta micro USB. Tendo como base o *datasheet* associado ao PAM2036, é possível dimensionar os valores dos condensadores e bobines presentes na imagem com o objetivo de conseguir as fontes de tensões já mencionadas. Os condensadores utilizados 10uF e 4.7uF, são valores standard impostos pelo *datasheet* associado existindo apenas uma variação nas bobines L1 e L2 obtidos através da equação [53]:

$$\Delta I_L = \frac{1}{(f)(L)} V_{OUT}(1 - \frac{V_{OUT}}{V_{IN}}) \tag{4.1}$$

Equation 4.1: Equação na obtenção das bobines.

Após a realização dos cálculos para valores de Vout 1.8 e 3.3 Volt, Vin 5 Volt e $\Delta I_L 1A$ o resultado obtido para ambos os casos foram duas bobines de 4.7 uH, como está representado. O regulador de tensão AP7115, semelhante ao PAM2306 é também capaz de suportar um valor de tensão de entrada presente no intervalo de valores entre 2.5 e 5.5 Volt. Para o circuito presente na imagem 4.2, foi utilizada como tensão de entrada 3.3 proveniente do regulador analisado anteriormente. O AP7115 tem como função produzir uma tensão de 2.5 Volt, recorrente para funcionalidades como a possibilidade de conectar o Compute Modules através de um *composite cable*. Os condensadores C4 e C9 recomendados por parte do *datasheet*, relativamente ao *output capacitator*, C5, este é variável sendo aconselhável um valor igual ou superior a 1uF [54].

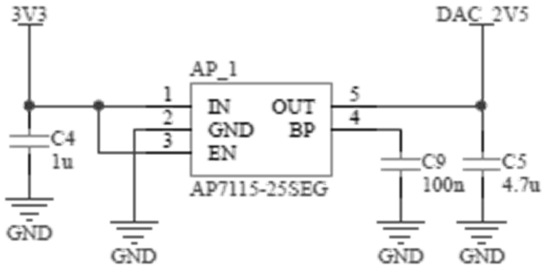


Figura 4.2: Esquemático relativo ao AP7115.

4.1.3 HDMI

O HDMI tem como objetivo possibilitar uma interação rápida entre o sistema e o utilizador, permitindo assim realizar configurações associadas de forma rápida e simples. Os dados são transmitidos a partir dos pinos de data TMDS (*Transition Minimised Differential Signalling*), do 1 ao 12 relativo ao conector HDMI, que por sua vez estão conectados aos pinos do CM3 tal como

esta representado na figura 4.3. As linhas SDA (Serial Data) e SCL (*Serial Clock*), protocolo de comunicação I2C, são utilizados na comunicação com o CM3.

Possuindo este conector um revestimento metálico, torna-se necessário proceder a medidas no que toca a proteções contra descargas elétricas. Para tal, é introduzido o integrado TPD12S016 a qual estão conectadas todas as linhas provenientes do conector HDMI bem como a alimentação do mesmo. A proteção é garantida recorrendo à utilização de díodos TVS, impedindo que tensões superiores à permitida possam danificar os componentes associados.

O sinal HPD (*Hot Plug Detected*) permite determinar quando o conector HDMI possui uma ficha conectada a si. Auxiliado de um MOSFET, M6, o sinal HPDF proveniente do integrado de proteção ESD é introduzido na gate de modo a produzir uma tensão na drain quando esta possui um valor superior a 0 volt, sendo garantido 1.8 Volt através da resistência de *pull up*. Como já foi mencionado, o conector possui um chassi metálico sendo assim necessário isolar a massa do mesmo. Este isolamento é realizado através da utilização de uma resistência e condensador, criando uma ponte entre o $HDMI\_AGND$ e GND, capaz de evitar possíveis ruídos. Os restantes condensadores utilizadas tal como as resistências, são impostos pelo *datasheet*.

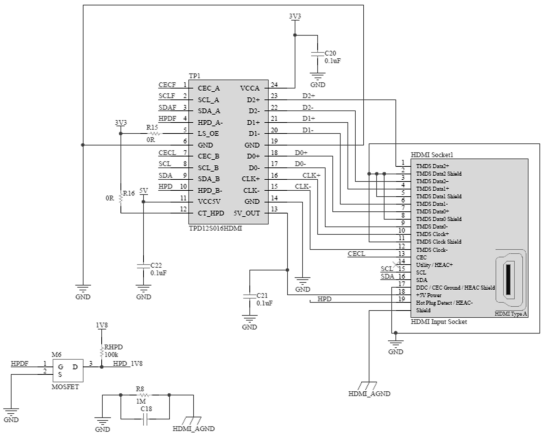


Figura 4.3: Esquemático relativo à secção HDMI.

Relativamente ao processo de design da PCB, é necessario ter em consideração regras associadas ao HDMI tendo como foco as linhas de dados provenientes do conector associado. Os pares diferencias (linhas de dados) devem possuir semelhanças como o comprimento e largura da pista de modo a diminuir os riscos de interferências Eletromagnéticas (EMI). No que toca ao encapsulamento, sendo o conector HDMI metálico, foi desenhada uma *shape*, de forma a abranger os pinos $AGND$ criando assim o isolamento já mencionado.

4.1.4 USB

O esquemático relativo aos conectores USB, aborda as três portas sendo duas destas do tipo A para realizar comunicação com periféricos, como teclados, *dongles* ou pens USB, e uma entrada

micro USB tendo como funcionalidade apenas a receção da imagem/sistema operativo. Como o CM3 possui apenas um par de linhas relativo ao sinal USB, revela-se necessário a obtenção de uma solução que permita a "expansão" do sinal de modo a utilização deste por parte dos conectores USB mencionados.

Sendo a porta micro USB utilizada apenas durante o processo de flash do CM3, esta não necessita do sinal USB ao longo do funcionamento normal da *board*, requisitando o sinal apenas durante o processo referido. Assim, foi utilizado o integrado FSUSB24MUX que se encontra equipado com uma entrada de sinal USB e duas saídas, igualmente, de sinal USB tendo como função retransmitir o sinal de entrada por um dos pares de pinos, 6 e 7 ou 8 e 9, consoante o valor presente no pino SEL. O integrado funciona como um simples *switch*, como esta representado na imagem 4.4, que reencaminha o sinal de entrada para $HSD1$ caso o $SEL = 0$ e para o $HSD2$ caso o $SEL = 1$. A linha $HSD1$ é referente ao micro USB, sendo a $HSD2$ destinada ao integrado LAN9512 que será abordado no seguinte subcapítulo.

Tal como o conector HDMI, os três conectores USB possuem também um chassi metálico o que conduz à necessidade de recorrer a proteções ESD. Os integrados USBLC6-2 e IP4220CZ6, possuem um funcionamento semelhante ao apresentado por parte do TPD baseado em díodos TVS sendo que cada uma das linhas de dados encontra-se protegida por estes integrados.

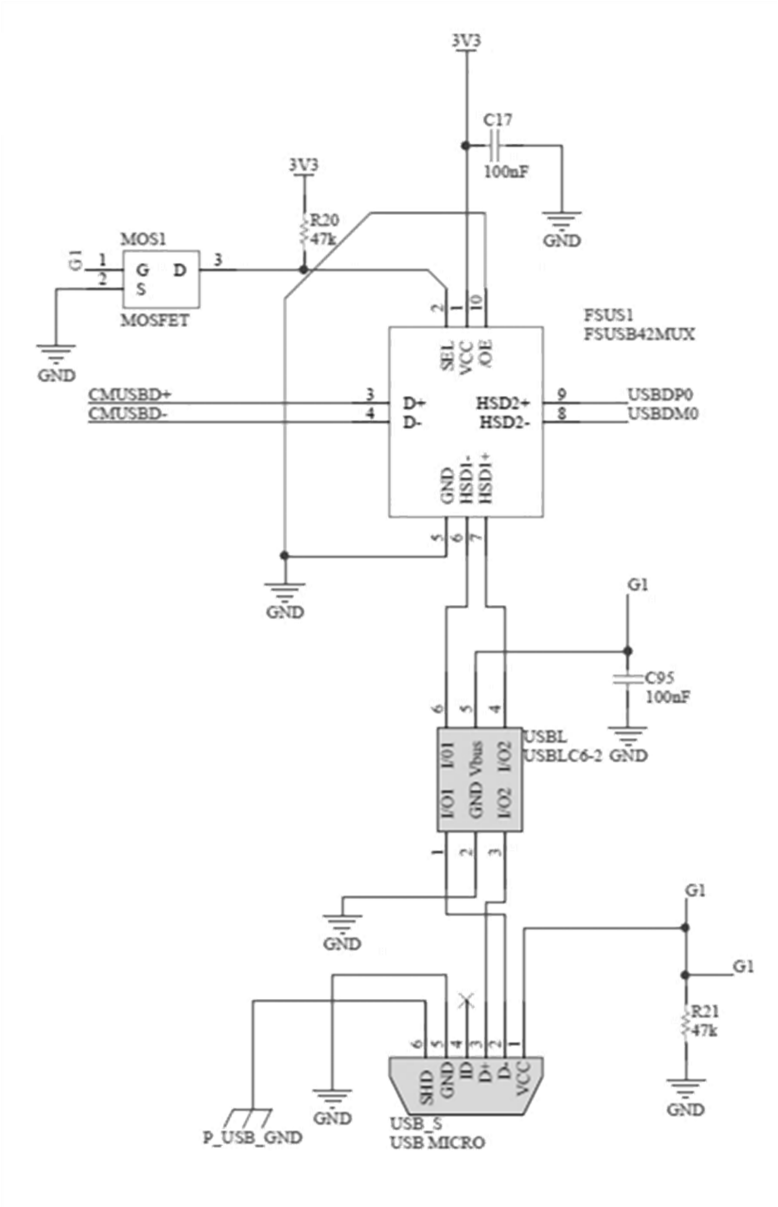


Figura 4.4: Esquemático representa secção da board relacionada com as portas USB e micro USB.

4.1.5 LAN

O parte do bloco LAN e USB, é dedicado inteiramente ao integrado LAN9512 e ao conector Ethernet. Como tem vindo a ser mencionado, o LAN9512 é responsável por garantir a comunicação com a porta Ethernet e as duas portas USB do tipo A. O integrado em questão possui a capacidade de gerar dois pares de sinais USB, $UDBDM2/USBDP2$ e $UDBDM3/USBDP3$, o que permite a utilização dos dois conectores associados, porém este necessita de um diferencial de entrada, $UDBDM0/USBDP0$, levando o bloco LAN estar ligado ao bloco USB. Como foi verificado no ponto anterior, o integrado $FSUSBMUX$ possui a capacidade de

reencaminhar um sinal USB dependendo da utilização da porta micro USB, assim caso esta não esteja em utilização o sinal é reencaminhado para o integrado LAN9512.

O LAN9512 necessita de um sinal estável de *clock*, o que leva o integrado a incorporar dois pinos de saída e entrada para este propósito, XO e XI. Este sinal pode ser obtido recorrendo a um oscilador externo com uma determinada frequência ou gerado recorrendo a um dos pinos da Raspberry Compute Module. Para o projeto em questão e tendo em consideração o *datasheet* do LAN9512, foi escolhida a primeira opção referida seguindo também a sugestão elaborada pelo fabricante. Assim, como está representado na imagem 4.6 foi escolhido um cristal de 25 MHz baseado no circuito sugerido pela Microchip (fabricante do LAN9512) tal como os condensadores em paralelo, com o intuito de ajudarem na estabilização da frequência, e a resistência de 1 MOhms para o funcionamento correto.

Relativamente à interação com o conector Ethernet, HR91105A, este é realizado recorrendo aos pinos de dados respetivos para receber e enviar informação: TXP, TXN, RXP, RXN. As resistências com valores de 49.9 Ohms (R41-R46) presentes nos terminais de cada pino do RJ45 permite preservar a impedância presente no cabo coaxial Ethernet, caso estas fossem descartadas a possibilidade de existir reflexões (envio de sinais para a origem) seriam elevadas. Este fenómeno é causado pelas diferencias de impedância, neste caso entre o cabo (50 Ohms) e as linhas destinas a este propósito, que podem danificar a informação transmitida. Os condensadores, por sua vez tem como função filtrar as frequências altas de modo a proporcionar um sinal estável em cada uma das linhas.

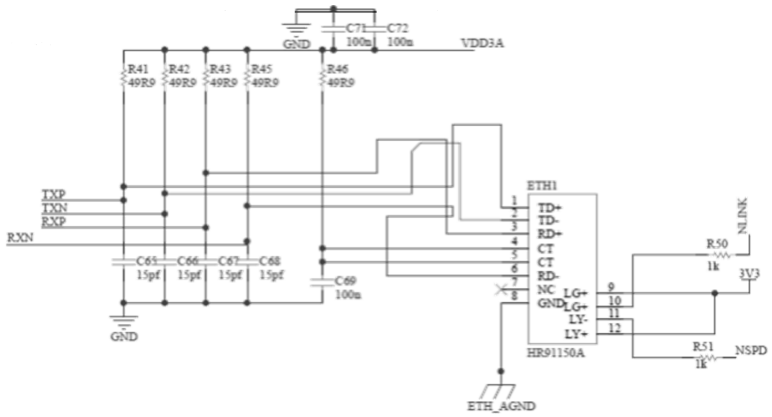


Figura 4.5: Esquemático relativo à porta Ethernet.

Ao longo do design da PCB relativo à fase de implementação do esquemático LAN, foi necessário ter em consideração algumas regras para o correto funcionamento do integrado LAN9512. Os condensadores relacionados com alimentação (*bypass capacitators* utilizados na filtragem de altas frequências), devem ser posicionados o mais próximo possível dos respetivos pinos do modulo e em todas as entradas associadas a alimentações.

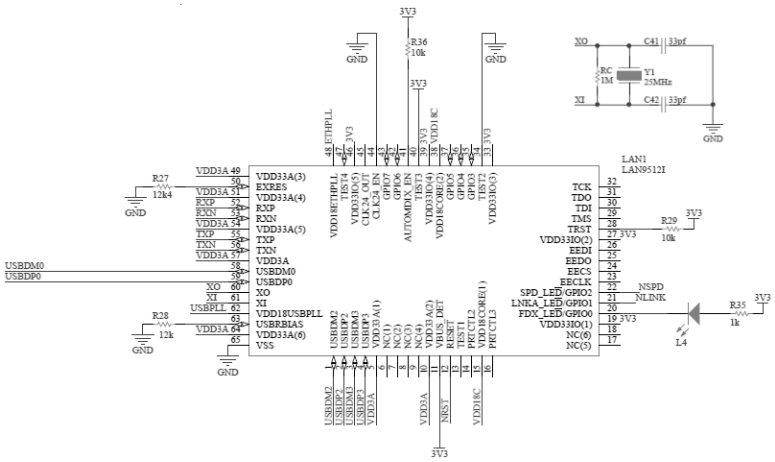


Figura 4.6: Esquemático relativo ao LAN9512.

O conector Ethernet relativamente ao design da PCB, possui regras e consequentemente procedimentos idênticos aos realizados nas portas USB mencionadas anteriormente. Possuindo também pares diferencias ($TRxP/TRxN$ sendo x=0-3) estes devem apresentar o mesmo comprimento e o mais próximo possível entre si, com um valor de 5 milis bem como um afastamento considerável de outros circuitos. Os pares diferenciais provenientes do conector RJ45, foram desenhados de modo a possuírem uma distância máxima possível entre si com o intuito de minimizar ruídos. No que toca ao $AGND$ do RJ45, foram tomadas as mesmas medidas já mencionadas para a situação apresentada por parte dos conectores USB e HDMI.

4.1.6 Resultado

Após o desenvolvimento de todo o hardware, o gateway encontra-se preparado para ser colocado no encapsulamento mencionado no capitulo anterior. Assim a imagem representa *board* obtida e o produto final pronto a ser posicionado numa rede LoRaWAN. Como é possível verificar através das figuras, para o encapsulamento foi considerado o posicionamento das antenas (LoRa e SIM800) no exterior da caixa sendo consequentemente realizado a devida proteção contra líquidos nos locais onde foram realizados os furos para o fim mencionado.

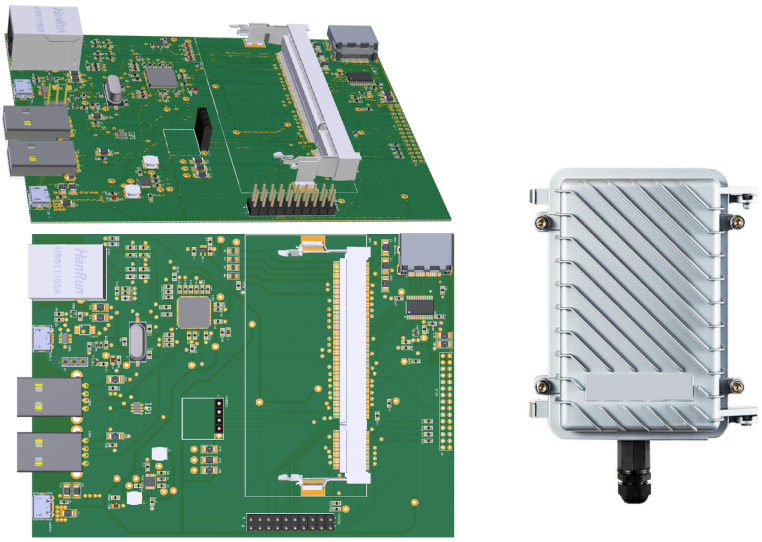


Figura 4.7: Representação da PCB final obtida e o encapsulamento selecionado para o prototipo final.

4.2 Software

Após uma análise do hardware desenvolvido, encontram-se reunidas as condições necessárias para descrever o software desenvolvido para a *Custom Board*. Este capítulo, tem assim como objetivo descrever as configurações realizadas para os periféricos utilizados e a implementação do algoritmo responsável pelo funcionamento principal do gateway. Além dos pontos mencionados, na última secção do capítulo é analisada a plataforma web desenvolvida utilizada na interpretação dos dados recolhidos.

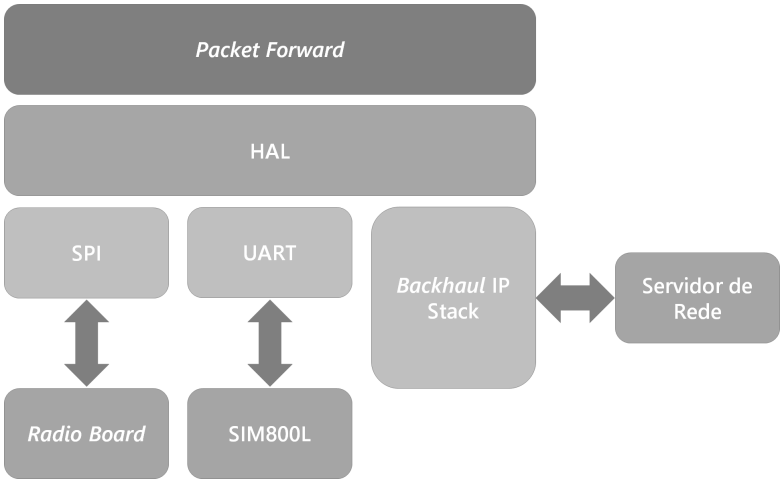


Figura 4.8: Representação da software stack.

A Figura 4.8 representa a *stack* referente à *Custom Board*, onde numa camada inferior estão presente os periféricos utilizados ao longo da dissertação, seguindo-se de uma camada intermédia representada pelo sistema operativo e terminando na camada aplicacional que por sua vez incorpora as bibliotecas e aplicações presentes na *board*.

4.2.1 Configurações

Previamente à realização da implementação referente ao software, todo o sistema necessita de percorrer uma fase de configurações com o intuito de fornecer o correto e desejado funcionamento pretendido. Maioritariamente as configurações mencionadas ao longo do presente subcapítulo, encontram-se relacionados com os periféricos utilizados, auxiliadas de uma descrição simples de modo a possibilitar o utilizador a realizar futuras mudanças no hardware. Além das configurações mencionadas, a TTN necessita também de ser integrada na rede. Assim, neste subcapítulo é descrito de que forma é realizada a configuração da *HTTP Integration* e como é implementado o *decoder* responsável pelo tratamento dos dados recebidos na TTN.

4.2.1.1 Sistema Operativo

O Sistema Operativo (OS) presente na *Custom Board* é o *Raspbian Stretch Lite* versão 4.14 fornecido pela Raspberry Pi. O processo de instalação do SO, requer a utilização da porta micro-usb com o intuito de realizar o flash da memória eMMC presente no CM3. Assim, foi utilizado um *installer* que permite a instalação das driver e da *boot tool*, num sistema Windows, de modo a reconhecer o CM3 e tornando-se assim possivel conectar a *Custom Board* ao sistema Windows. Posteriormente, é utilizado o executável rpiboot (*boot tool*) e o Win32DiskImager para escrever a imagem, concluindo assim o procedimento de flash.

4.2.1.2 Overlays

Como já mencionado nas especificações do sistema e tendo por base o *datasheet* referente ao Raspberry Compute Module, é de notar que este micrcontrolador encontra-se equipado com elevado número de GPIOS permitindo a conexão de uma elevada quantidade de periféricos. Porém, estes GPIOs possuem, geralmente, quatro funcionalidades associadas sendo por defeito selecionada a primeira, como revela a seguinte tabela associada aos GPIOs utilizados pela *Costum Board* na interface com o ENC28J60 e SIM800.

Tabela 4.1: Funções dos GPIOs utilizadas para comunicar com os periféricos ENC28J60 e SIM800.

GPIO	ALT0	ALT1	ALT2	ALT3	ALT4
18	PCM_CLK	SD10	-	-	SPI1_CE0_N
19	PCM_FS	SD11	-	-	SPI1_MISO
20	PCM_DIN	SD12	-	-	SPI1_MOSI
21	PCM_DOUT	SD13	-	-	SPI1_SCLK
32	GPCLK0	SA1	-	TXD0	-
33	FL1	SA0	-	RXD0	-

De modo a possibilitar as funções alternáveis dos GPIOs, o *kernel* fornece *overlays* e guias vocacionados para desenvolvimento de *overlays* adicionais que não estão presentes no sistema por defeito. A descrição do hardware presente na *Costum Board* é realizado recorrendo a uma *device tree* responsável por identificar e executar as drivers associadas a um determinado hardware, sendo estes componentes identificados pelos nós presentes na *device tree*. Porém, nem todos os periféricos estão presentes na *device tree* como é o caso do ENC28J60, o que conduz à necessidade da introdução destes nos respetivos nós. De modo a resolver o problema, o *Kernel* possibilita a criação de *overlays* que tem como função permitir a inserção de novos dispositivos, não reconhecidos, na *device tree*. A adição de *overlays* resulta assim no reconhecimento de hardware novo que é processado numa das etapas do *boot* [55].

A seguinte secção de código, representa o *overlay* exemplo que possibilita a utilização do ENC28j60. Como é possível analisar, o *overlay* é dividido em dois *fragments* e um *override*, sendo o primeiro destinado ao nó e o ultimo à utilização de parâmetros por parte do nó. O primeiro *fragment* em questão, *fragment0*, possui um *target* que identifica o nó a qual é aplicado o *overlay*, sendo para o ENC28J60 o SPI2 seguindo-se o *__overlay__* que representa a informação a ser inserida no nó referenciado pelo *target*. Ainda no *fragment0*, o *overlay* deste é destinado ao periférico *eth1* sendo para o caso em específico o ECN28J60. A partir da *label reg* é possível identificar qual dos 3 *slaves* (0,1 ou 2) é utilizado por parte do SPI1 e como já mencionado na especificações do sistema será o 0. As *labels interrupt* e *spi-frequency* permitem configurar a interrupção requisitada por parte do pino INT do modulo ENC28j60 e a frequência do SPI1 respetivamente. Relativamente ao *fragment1*, este tem como *target* o gpio tendo como objetivo atribuir um novo pino ao conjunto *eth1_pins*, sendo este o GPIO 0 destinado à interrupção mencionada anteriormente. A secção *__overrides__* e que define a conclusão do ficheiro *overlay*, permite realizar a passagem de parâmetros, definidos anteriormente, e os quais estão presentes no nó SPI1 da *device tree*.

```
1 {
2 compatible = "brcm,bcm2708";
```

```
3
4   fragment@0 {
5     target = <&spi1>;
6     __overlay__ {
7     /* needed to avoid dtc warning */
8       #address-cells = <1>;
9       #size-cells = <0>;
10      status = "okay";
11
12      eth1: enc28j60@0{
13        compatible = "microchip,enc28j60";
14        reg = <0>; /* CE0 */
15        pinctrl-names = "default";
16        pinctrl-0 = <&eth1_pins>;
17        interrupt-parent = <&gpio>;
18        interrupts = <25 0x2>; /* falling edge */
19        spi-max-frequency = <12000000>;
20        status = "okay";
21      };
22    };
23  };
24
25  fragment@1 {
26    target = <&gpio>;
27    __overlay__ {
28      eth1_pins: eth1_pins {
29        brcm,pins = <25>;
30        brcm,function = <0>; /* in */
31        brcm,pull = <0>; /* none */
32      };
33    };
34  };
35
36  __overrides__ {
37    int_pin = <&eth1>, "interrupts:0",
38    <&eth1_pins>, "brcm,pins:0";
39    speed = <&eth1>, "spi-max-frequency:0";
40  };
41 };
```

Listing 4.1: Exemplo do *overlay* utilizado no ECN28J60

Após completado o ficheiro de extensão .dts que contém o *overlay* descrito, é necessário compilar este de modo a obter o respetivo código objeto de extensão .dtbo:

```
1 dtc -@ -I dts -O dtb -o enc28j60-spi1.dtbo enc28j60-spi1-overlay.dts
```

Tal como o ENC28J60, o SIM800 requer também a utilização de um *overlay* com o objetivo de ativar a UART0 nos GPIOs mencionados anteriormente. Porém, não é necessario a criação de um *overlay* novo devido a este já estar presente por defeito sendo apenas necessario editar o ficheiro *config.txt*. A seguinte imagem representa a secção modificada no ficheiro config.txt, incluindo os *overlays* mencionados e adicionais requisitados no funcionamento do periférico SPI1.

4.2.1.3 TTN Http Integration

Tal como foi mencionado no capítulo anterior, foi utilizada a *HTTP Integration* fornecida por parte da plataforma TTN. Assim, é necessario descrever quais as configurações associadas a esta *integrations* que tem como objetivo reencaminhar toda a informação, transmitida pelo nós, para a aplicação web desenvolvida.

Assim, recorrendo à consola TTN é possível optar pela *HTTP Integration* onde posteriormente é escolhido o método POST e o URL de destino. Após a realização da etapa referida, é desenvolvido um algoritmo de *decoder*, apresentado na seguinte secção, recorrendo novamente à consola TTN. Este breve algoritmo desenvolvido em JavaScript, permite interpretar os pacotes recebidos e enviar estes com o formato apresentado no *listining*, com o intuito de fornecer toda a informação devidamente tratada. A implementação passa por realizar operações *bitwise* recorrendo a um *array*, bytes, no qual esta presente o *payload* em formato hexadecimal.

MAC Payload (xxbytes)		
12 bits	12 bits	12 bits
Temperatura	Humidade	Luminosidade

Figura 4.9: Formato do MAC *Payload* recebido pela TTN, a partir do qual é realizado o *decoder* descrito.

```
function Decoder(bytes, port) {
  var humidity = (bytes[0]<<4) | ((bytes[1] & 0xF0) >> 4);
  var temp = ((bytes[1] & 0x0F) << 8) | bytes[2];
  var lig = (bytes[3]<<4) | ((bytes[4] & 0xF0) >> 4);

  return {
    humidity: humidity/10,
    temp:(temp-1000)/10,
    ligth:lig/100
  }
}
```

Listing 4.2: Decoder da TTN

4.2.2 *Packet Forwarder*

A função fulcral do gateway desenvolvido é realizar o reencaminhamento de pacotes recebidos por diversos nós para o servidor de rede. Tendo em conta a análise realizada ao longo do capítulo anterior, é de realçar que a receção dos dados são obtidos recorrendo aos módulos

SX1275 e SX1301, sendo este ultimo responsável por estabelecer comunicação com o Raspberry Compute Module. De modo a facilitar a interface mencionada, a Semtech desenvolveu uma HAL, *open-source*, com a capacidade de configurar, ler ou escrever no base *band processor* SX1301, tendo como base o protocolo de comunicação SPI. Auxiliada desta HAL, a empresa em questão fornece também uma aplicação base que consiste em realizar uma leitura da FIFO, presente no SX1301, onde estão armazenados os pacotes recebidos pelos módulos de radio SX1275 e que posteriormente envia os dados para um servidor de rede. O programa em questão, permite também receber pacotes por parte do servidor com o objetivo de reencaminha-los para os *nós* presentes na rede, porém não é um funcionalidade, atualmente, relevante para o projeto em questão. [56]

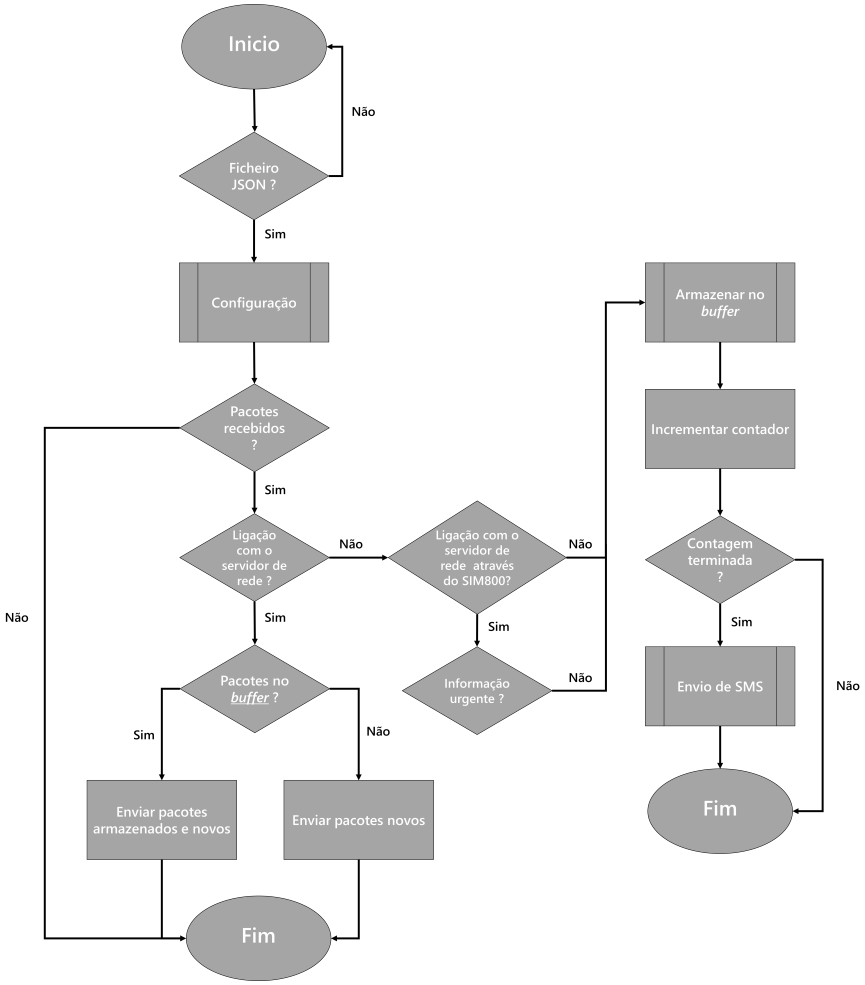


Figura 4.10: *Flowchart* com a representação da aplicação final presente no gateway obtido.

Contudo a aplicação em questão não contempla os requisitos mencionados como redundância nas comunicações ou o *backup* de pacotes para situações onde não existe comunicação com o servidor de rede, sendo necessario proceder a essas alterações. Assim, o subcapítulo em questão tem como objetivo analisar a aplicação final obtida como a representa no *flowchart* 4.10.

Numa primeira etapa, o gateway é configurado através de um ficheiro JSON onde esta presente o ID do mesmo, os canais ativos com as respetivas frequências e *spreading factors*, o servidor de rede destino e contacto do responsável da rede. Após realizada a inicialização do gateway, este encontra-se preparado para receber e reencaminhar os pacotes recebidos por parte dos nós.

A *Costum Board* realiza a leitura dos pacotes presentes no SX1301 num intervalo de tempo determinado por parte do utilizador. Estes são posteriormente armazenados com o intuito de serem enviados em formato JSON para o servidor de rede. Porém, quando a comunicação para com o servidor não é possível, os dados mencionados são perdidos e consequentemente é impossibilitada a devida monitorização da rede. Para tal, foi implementado um *buffer* circular com a capacidade de armazenar 200 pacotes com o intuito de evitar a menor perda possível de pacotes. Assim sempre que a ligação com o servidor for perdida, os pacotes são armazenados e enviados quando a ligação for novamente estabelecida.

Porém a ligação para com o servidor nem sempre é estabelecida novamente sendo este problema conduzido maioritariamente por falhas na cablagem (Ethernet) ou no periférico em si. Para estas situações, foi atribuído ao modulo SIM800 a função de enviar uma SMS, para o utilizador da rede, com o objetivo de este ser alertado da situação aqui descrita. Deste modo, através da utilização de comandos AT e recorrendo à interface UART, foi implementada uma biblioteca tendo como função realizar o envio de SMS. O ficheiro JSON de configuração mencionado anteriormente, possui também o numero do destinatário responsável por receber a informação do estado da ligação e o conteúdo da SMS a ser enviada. Esta mensagem apenas é enviada quando a ligação falhar um determinado numero de vezes, sendo este valor definido pelo utilizador através do ficheiro JSON mencionado.

```
1 set_get_cmd(fd, RW, "AT\r\n");
2 set_get_cmd(fd, RW, "AT+CMGF=1\r\n");
3 set_get_cmd(fd, RW, "AT+CNMI=2,1,0,0,0\r\n");
4 set_get_cmd(fd, RW, "AT+CMGS=\"961001620\"\r\n"); //Numero do utilizador
5 set_get_cmd(fd, WO, "Mensagem\r\n"); //Mensagem a enviar
6 set_get_cmd(fd, WO, "\x1A");
```

Listing 4.3: Sequencia de comando AT para o envio de SMS

```
1 set_get_cmd(fd, RW, "AT+CREG?\r\n");
2 set_get_cmd(fd, RW, "AT+SAPBR=3,1,\"Contype\",\"GPRS\"\r\n");
3 set_get_cmd(fd, RW, "AT+SAPBR=3,1,\"APN\",\"internetm2m\"\r\n");
4 set_get_cmd(fd, RW, "AT+SAPBR=1,1\r\n");
5 set_get_cmd(fd, RW, "AT+HTTPINIT\r\n");
6 set_get_cmd(fd, RW, "AT+HTTPPARA=\"CID\",1\r\n");
7 set_get_cmd(fd, RW, "AT+HTTPPARA=\"xxxx\",\"URL\"\r\n"); //URL destino
```

```
8  set_get_cmd(fd, RW, "AT+HTTPPARA=\"CONTENT\",\"application/json\"\r\n");
9  set_get_cmd(fd, RW, "AT+HTTPDATA=\r\n");
10 set_get_cmd(fd, RW, "sizeof(buff_send)\r\n"); //Tamanho mensagem
11 set_get_cmd(fd, RW, ",10000\r\n");
12 set_get_cmd(fd, RW, "buff_send\r\n"); //Conteudo mensagem
13 set_get_cmd(fd, RW, "AT+HTTPSSL=0\r\n");
14 set_get_cmd(fd, RW, "AT+HTTPACTION=1\r\n");
15 set_get_cmd(fd, RW, "AT+HTTPREAD\r\n");
16 set_get_cmd(fd, RW, "AT+HTTPTERM\r\n");
17 set_get_cmd(fd, RW, "AT+SAPBR=0,1\r\n");
```

Listing 4.4: Sequencia de comando AT para o envio de POST

Todos os pacotes reencaminhados pelo gateway são abstratos a este, ou seja, o gateway é impossibilitado de ter acesso ao conteúdo dos pacotes recebidos por questões de segurança, que o impedem de realizar encriptação ou desencriptação das mensagens. Porém, existe a possibilidade de receber pacotes com conteúdo urgente, indícios de incêndio, durante fases onde não há comunicação com o servidor de rede o que conduz a um alerta não antecipado, impedindo a rede de atuar devidamente. A resolução do problema apresentado, passa por reservar um dos oito canais presentes na *Radio Board*, foi escolhido um canal com uma frequência de 868.645 MHz, apenas para mensagens que possuem um conteúdo urgente de modo a informar o gateway qual o carácter da mensagem. Posteriormente, este pacote é enviado recorrendo ao módulo SIM800 através do método POST para o servidor de rede. Assim foi adicionada à biblioteca SIM800, as funções respetivas para possibilitar o modulo de comunicar com o servidor, tendo como base comandos AT semelhantes aos utilizados no envio de SMS.

Deste modo, a aplicação final obtida encontra-se representada no *flowchart* apresentado inicialmente, onde é possível verificar cada uma das fases descritas anteriormente. O capítulo seguinte, aborda os testes realizados ao funcionamento do sistema debruçando-se sobre a aplicação descrita de modo a comprovar o correto funcionamento da mesma.

4.2.3 Web Platform

A aplicação web representa a última etapa relativamente à implementação de software sendo uma secção mais abstraída do conteúdo presente na *board* desenvolvida. Como foi referenciado inicialmente, esta dissertação enquadra-se num projeto de maior escala que pretende prevenir incêndios florestais recorrendo a dispositivos com sensores adequados a esse fim que posteriormente enviam a informação recolhida para um gateway e este ultimo reencaminha os dados para um servidor de rede. Contudo, e tendo em consideração o âmbito do projeto, revela-se necessário interpretar estes dados de uma forma simples e representá-los graficamente de modo a garantir

uma compreensão fácil por parte do utilizador. Assim a aplicação web torna possível este processo recorrendo à representação geográfica do posicionamento dos nós alertando qual o local especifico na ocorrência de um possível incêndio.

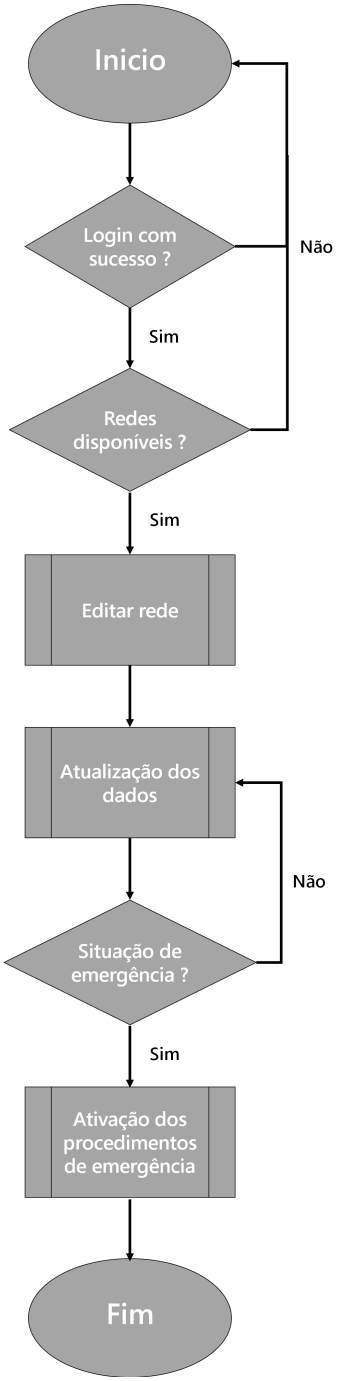


Figura 4.11: *Flowchart* das primeira duas etapas da plataforma web Login e Redes.

A aplicação encontra-se divida em três fases distintas, sendo que numa primeira etapa o utilizador realiza o login que é posteriormente verificado recorrendo à base de dados mencionada,

fornecendo assim acesso à segunda fase da aplicação. Após a validação das credenciais introduzidas, são mostradas as diferentes redes disponíveis, existindo a possibilidade de criar uma nova ou eliminar as existentes. Toda esta fase, tem como principal foco a utilização de *queries* com o objetivo de retirar a informação associado a um determinado utilizador. A seguinte imagem representa graficamente as duas fases mencionadas.

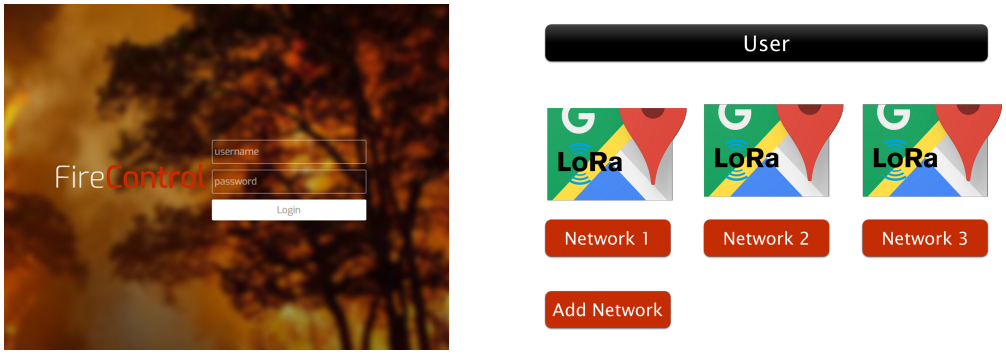


Figura 4.12: As primeira duas etapas da plataforma web Login e Redes.

Finalizado o processo de escolha ou edição das redes disponíveis, é garantido ao utilizador o acesso à rede escolhida. Recorrendo à API *Google Maps*, é possível verificar o posicionamento dos nós e gateways da rede, a introdução destes componentes é realizada manualmente recorrendo à latitude, longitude e ID associado. A remoção dos mesmos torna-se também possível devido ao *bind click*, fornecido pela API, que resulta futuramente numa alteração da base de dados eliminando o componente escolhido. Quando a rede é restaurada, a partir de uma sessão anterior, os componentes são automaticamente posicionados e as tabelas são preenchidas com os respetivos valores. Estes processos mencionados são obtidos recorrendo a métodos POST e ao AJAX (Assyncronous Javascript and XML) capaz de enviar dados para um ficheiro php que possuem como função enviar uma mensagem de formato JSON para a base de dados SQL com o intuito de realizar as tarefas acima descrita tal como representa o *listing* de um destes processos.

```
1  for(var key in markers)
2  {
3    myJSON.push({
4    "Type":"Node",
5    "ID":key,
6    "Lat":markers[key].position.lat(),
7    "Lng":markers[key].position.lng()
8    })
9  }
10
11 var myJSONP = JSON.stringify(myJSON);
12
```

```
13 for(var key1 in gates)
14 {
15   myJSONG.push({
16   "Type":"Gate",
17   "ID":key1,
18   "Lat":gates[key1].position.lat(),
19   "Lng":gates[key1].position.lng()
20   })
21 }
22
23 var myJSONP1 = JSON.stringify(myJSONG);
24
25 $.ajax ({
26   type: 'POST',
27   url:'go.php',
28   data: {'nodes' : myJSONP,
29   'gates': myJSONP1} ,
30   success: function(response) {
31   alert('Inserted!');
32   }
33 });
```

Listing 4.5: Exemplo de um POST através do AJAX

A Figura 4.13 remete para o *layout* da aplicação, onde é possível distinguir as diferentes secções assinaladas. A tabela tem como intuito representar os dados recebidos pelos diversos nós e alertar para qualquer irregularidade, auxiliada de um aviso presente no mapa. Desta forma as unidades de combate a incêndios, tem a possibilidade de receber um alerta em tempo real e com a localização exata da ocorrência, obtendo assim uma gestão simples e fácil relativamente a uma determinada área florestal.

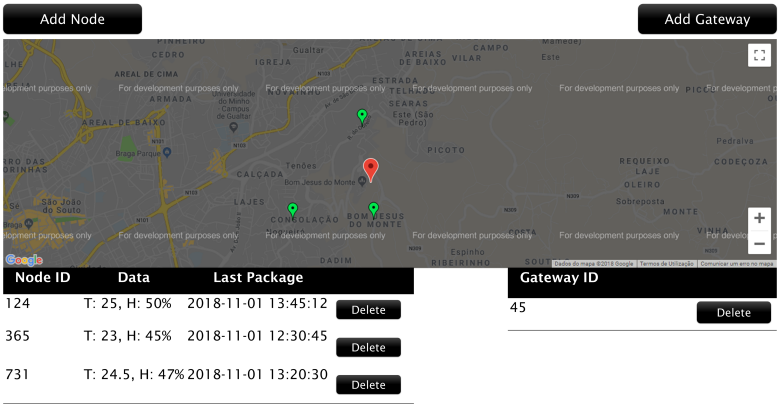


Figura 4.13: Após escolhida a rede, o utilizador é presentado com a interface gráfica aqui representada.

4.3 Conclusão

Este capítulo teve como intuito apresentar as diversas tarefas realizadas na obtenção do sistema final. A secção de hardware descreve detalhadamente de que modo foi obtida a *Custom Board*, seguindo-se do software respetivo que permite ao gateway realizar as funções pré definidas. Ainda relacionado com software, foi apresentado o desenvolvimento da aplicação web destinada ao utilizador da rede tendo como objetivo interpretar os dados recebidos pelos gateway.

Durante todo este processo, de implementação, foram encontradas algumas restrições. Relativamente ao hardware, apenas foi possivel realizar o design da PCB recorrendo a duas camadas e os componentes selecionados foram restringidos nas dimensões e stock do fornecedor.

Capítulo 5

Testes e Resultados

O presente capítulo tem como principal objetivo analisar os testes realizados tal como os resultados obtidos provenientes dos mesmos. Os testes aqui apresentados, visam principalmente o comportamento do gateway quando posicionado numa rede incorporada com os restantes componentes, servidor de rede e nós. São também realizados testes mais específicos, sendo estes direcionados ao comportamento do software de modo a verificar o resultado obtido após a implementação das modificações inseridas no algoritmo fornecido por parte da Semtech. Ainda neste capítulo, são mencionados alguns dos testes realizados antes da obtenção do protótipo final e à aplicação web desenvolvida.

Como foi mencionado, uma grande parte dos testes debruça-se sobre varieis como tempos e distancias sendo os restantes focados na secção de software desenvolvido. Assim os próximos subcapítulos apresentam em maior detalhe os testes desenvolvidos após a obtenção do *Custom Board* final.

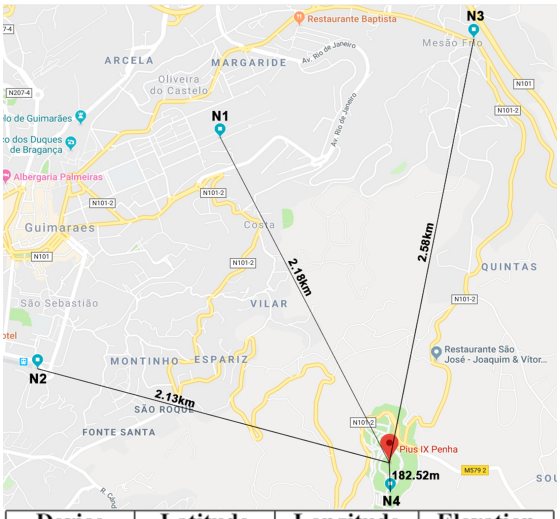
5.1 Testes de alcance

Umas das principais funcionalidades do gateway desenvolvido é conseguir realizar o reencaminhamento de todas as mensagens recebidas pelos diversos nós da rede. Porém, nem todos estes pacotes são recebidos com sucesso por parte do gateway em consequência de interferências, distancias elevadas e obstáculos presentes entre os *end-devices* e gateway. Assim, revela-se fulcral determinar em que condições são verificadas as falhas na receção dos pacotes consoante o posicionamento dos dispositivos (distancia e altura).

Deste modo, os testes abordados no presente subcapítulo possui como objetivo posicionar o gateway num local estratégico, altitude elevada, e enviar para o mesmo pacotes provenientes de um *end-device* com posição variável. Com o intuito de obter melhor resultados, foram realizados dois testes em locais diferentes como é possível verificar através das figuras 5.1 e 5.3.

Num primeiro teste, para cada posição do *end-device*, foram enviados pacotes com *data rates* diferentes, ou seja, para cada SF (SF7-SF12) foram enviados dois pacotes através da variação do *transmission power* (TxP1-TxP5). A alteração de parâmetros para cada uma das posições,

permite assim obter uma analise com maior exatidão acerca do comportamento dos dispositivos da rede.



Device	Latitude	Longitude	Elevation
Node 1	41,447191	-8,280558	215 m
Node 2	41,434758	-8,293337	208 m
Node 3	41,452558	-8,262819	331 m
Node 4	41,428094	-8,268661	590 m
Gateway	41,429672	-8,26854	603 m

Figura 5.1: Posicionamento dos nós e gateway ao longos dos testes realizados na Penha.

A seguinte tabela é referente aos testes realizados na zona de Guimarães, onde o gateway foi posicionado na Penha como representa a figura 5.1. Cada uma das colunas apresenta a percentagem de mensagens recebidas nas diferentes posições consoante o SF escolhido.

Tabela 5.1: Percentagem de pacotes recebidos em cada um dos nós.

Node	SF7	SF8	SF9	SF10	SF11	SF12
N1	0%	3,3%	16,6%	30%	46,7%	43,3%
N2	93,3%	50%	60%	50%	50%	66,7%
N3	0%	0%	0%	0%	0%	0%
N4	3,3%	13,3%	26,7%	30%	30%	43,3%

O gateway foi posicionado no ponto mais alto em relação aos nós. Para a posição N1, grande parte das mensagens falha para valores abaixo de SF10 sendo a maior causa destas falhas a presença de obstáculos e a não total linha de vista para com o gateway. Porém para a posição N2 é verificada a situação oposta, grande parte das mensagens são corretamente recebidas verificando-se para este caso a presença de poucos obstáculos e linha de vista para com o gateway. O caso N3 apresenta a maior distância entre os dispositivo e o gateway comparativamente aos outros casos. Este fator bem como a não linha de vista com o gateway, devido à presença de edifícios, conduz a uma total falha na receção das mensagens em cada um dos testes. Por último

a posição N4, revelou resultados pouco expectáveis. Este é o caso que apresenta menor distância, contudo a presença de obstáculos (rochas de grande dimensão) e possíveis interferências conduzem aos resultados obtidos.

Tabela 5.2: Distância e linha de vista entre as diversas posições e o gateway.

	N1	**N2**	**N3**	**N4**
Distância	2.18 km	2.13 km	2.58 km	185.52 m
Linha de vista	Não	Sim	Não	Não

Para obter uma melhor compreensão dos resultados obtidos, foi também realizado o levantamento dos valores de RSSI e SNR para os casos em cima observados. O SNR encontra-se representado pelo eixo vertical, sendo o eixo horizontal referente ao RSSI. O gráfico a) representa os valores de RSSI e SNR para o primeiro teste realizado, N1, onde os valores obtidos são bastante dispersos, porém, de acordo com o RSSI, estes, encontram-se fora da zona de ruído (-120 dBm). Por outro lado, através do SNR é de notar que a qualidade do sinal em diversos dos pacotes recebidos apresenta valores inferiores a 0 e perto da zona de ruído (-20 dB), sendo para algumas destas situações obtido um sinal fraco.

Relativamente ao gráfico b), os valores de SNR como os de RSSI apresentam resultados satisfatórios, evidenciando os de SNR com diversas situações onde os valores são superiores a 0. O resultado obtido no gráfico c), apresenta valores semelhantes aos representados no primeiro gráfico, a), sendo assim o segundo caso aquele que possui na totalidade os melhores resultados.

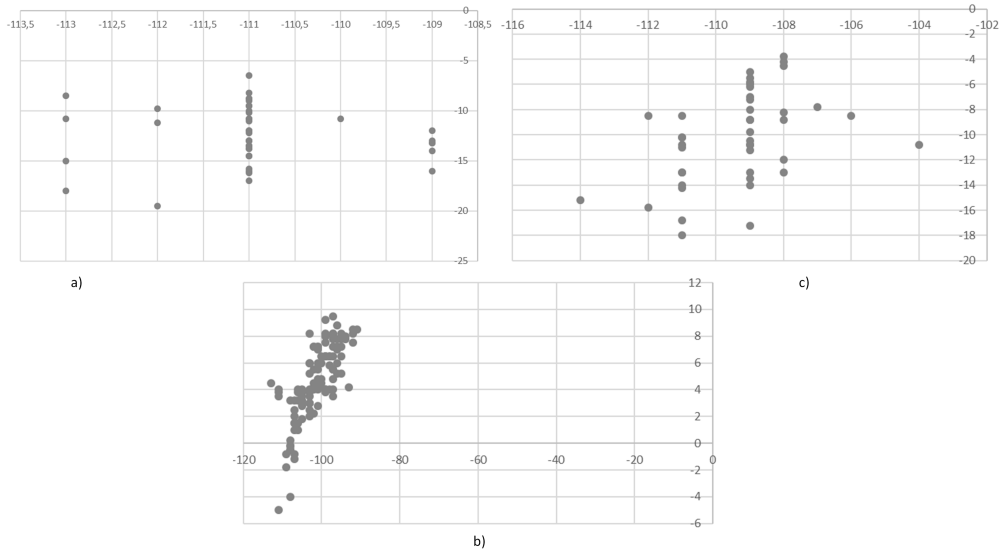
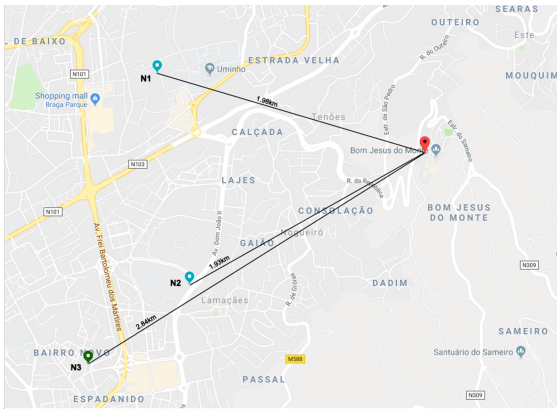


Figura 5.2: Gráficos com os valores RSSI (eixo horizontal) e SNR (eixo vertical) para cada uma das posições dos testes realizados na Penha.

O último teste foi realizado no distrito de Braga, tendo o gateway como localização o Monte Bom Jesus. Para estes testes foram utilizados dois nós sendo um destes estacionário e o restante

utilizado em duas localizações diferentes, como está representado na seguinte figura. Novamente, foi utilizado um meio urbano neste segundo teste.



Device	Latitude	Longitude	Elevation
Node 1	41,560078	-8,400911	189 m
Node 2	41,546564	-8,398164	205 m
Node 3	41,541495	-8,406812	203 m
Gateway	41,555111	-8,378013	381 m

Figura 5.3: Posicionamento dos nós e gateway ao longos dos testes realizados no distrito de Braga.

O método utilizado nestes testes foi semelhante ao teste anterior recorrendo as variações de SF e TxP. Para este caso, os valores de SF foram variados desde SF7 até SF12 sendo que para TxP foram utilizados apenas três: TxP1, TxP3 e TxP5. Assim, para cada valor de SF, começando em SF12 para SF7, cem mensagens foram enviadas através de uma combinação dos valores de SF com os três TxP escolhidos. Esta combinação foi utilizada nos testes de N1 e N2 resultando assim num total de mil e oitocentos pacotes por teste. O nó estacionário por outro lado, encontrava-se constantemente a transmitir realizando também a combinação de valores mencionada.

Tabela 5.3: Percentagem de pacotes recebidos em cada um dos nós.

Node	SF7	SF8	SF9	SF10	SF11	SF12
N1	85%	90%	100%	100%	100%	100%
N2	80%	85%	90%	100%	100%	100%
N3	0%	0%	0%	0%	0%	0%

A posição escolhida para o gateway foi o Bom Jesus, porém este não é o local que possui mais altitude no distrito de Braga. O Sameiro representa o ponto mais elevado da localidade tendo sido numa primeira análise o ponto escolhido para o gateway, contudo este local possui um úmero elevado de antenas que podem provocar interferências na receção das mensagens, acabando por descartar este local para o gateway. A escolha do local teve também influencia dos primeiros testes, o posicionamento do gateway na Penha sofreu possíveis interferências de antenas vizinhas provocando as percentagens mencionadas. Assim, tornou-se relevante realizar

novos testes de alcance em condições diferentes, tendo sido verificados resultados melhores em relação aos testes realizados na Penha.

Para a posição N1, o nó foi posicionado num parque de estacionamento com linha de vista para o gateway, sendo este o teste com maior proximidade do gateway. Maior parte das mensagens foram recebidas com sucesso, realçando as que apresentam um menores *data rates* (valores de SF maiores). Já expectável devido à longa distância em zona urbana, para valores de SF menores algumas das mensagens não foram recebidas.

Na segunda posição (N2) o nó foi também colocado num parque de estacionamento com linha de vista para com o gateway sendo a elevação um pouco superior de N1. O resultados obtidos foram semelhantes ao de N1, contudo devido a uma distancia superior em relação a este, para os valores de SF7, SF8 e SF9 notou-se uma ligeira perda de mensagens conforme seria expectável devido à distancia elevada.

Para a posição N3, o nó foi posicionado no quinto andar de um prédio com linha de vista para o gateway apresentando a maior distância das três posições. Este nó encontrava-se a transmitir constantemente variando os parâmetros de acordo com as combinações referidas. Comparando os resultados obtidos com os restantes testes, este não recebeu nenhuma mensagem com sucesso concluindo assim que a distancia é a principal causa dos resultados obtidos.

Tabela 5.4: Distância e linha de vista entre as diversas posições e o gateway.

	N1	**N2**	**N3**
Distância	1.93 km	1.98 km	2.85 km
Linha de vista	Sim	Sim	Sim

Tal como foi apresentado no primeiro teste, para este foi também realizado o levantamento dos valores de RSSI e SNR para as duas posições N1 e N2. O gráfico b) representa os valores de RSSI e SNR obtidos na posição N1, onde os valores são muito semelhantes nunca atingindo a zona de ruído (-120 dBm). Quanto ao SNR, para alguns dos casos a qualidade do sinal é fraca sendo estes causados pelo elevado *data rate*. O gráfico a) por sua vez representa os resultados SNR e RSSI para a posição N2, onde os valores obtidos são semelhantes ao de N1. Contudo em N2, a qualidade do sinal é inferior e mais aproximada da zona de ruído.

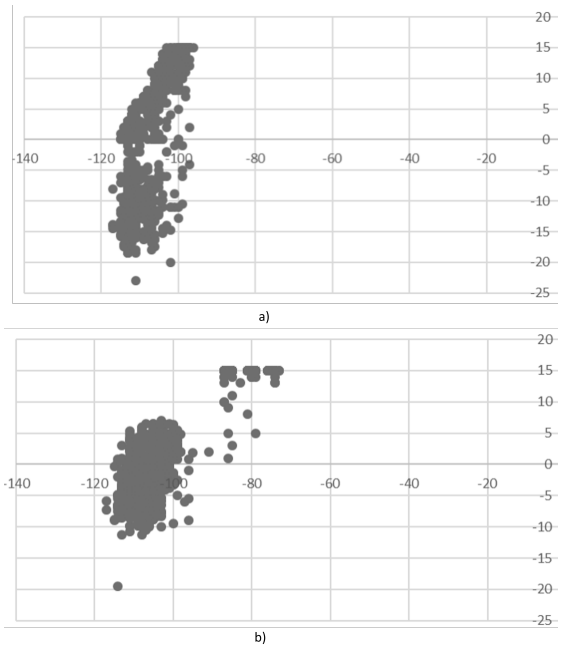


Figura 5.4: Gráficos com os valores RSSI (eixo horizontal) e SNR (eixo vertical) para cada uma das posições dos testes realizados em Braga.

Para cada um dos testes, foi respeitado o mesmo formato de mensagem. O pacote proveniente do dispositivo contém assim: o id da aplicação destino, o id do nó, o numero serie do dispositivo e o payload respetivo. Posteriormente esta informação é armazenada num ficheiro JSON, onde são adicionados campos, metadata, referente ao gateway recetor do pacote.O seguinte *listing*, representa o ficheiro JSON final a enviar para o servidor de rede, constituído pelas diversas mensagens recebidas. Como todo o payload é abstrato ao gateway, (informação encriptada) o seguinte subcapítulo aborda a plataforma web desenvolvida e consequentemente quais os valores adquiridos pelos sensores ou seja o campo payload presente no ficheiro JSON.

```
{
"app_id":"1323232321",
"dev_id":"1223",
"hardware_serial":"0000000000000109",
"port":5,
"counter":55,
"payload_raw":"AAAA",
"metadata":
  {
    "time":"2018-11-16T17:16:46.214146755Z",
    "frequency":868.3,
    "modulation":"LORA",
    "data_rate":"SF12BW125",
    "coding_rate":"4/5",
    "gateways":
      [
        {
          "gtw_id":"eui-aa555a0402230101",
          "timestamp":488802892,
          "time":"",
          "channel":1,
          "rssi":-109,
          "snr":-8,
          "rf_chain":1,
          "latitude":41.551163,
          "longitude":-8.375217,
          "location_source":"registry"
        }
      ]
  },
"downlink_url":"https://integrations.thethingsnetwork.org/ttn-eu/api/v2/down
/1323232321/1243?key=ttn-account-v2.KqQRkF63WDF2_VmHRrA8QiI8h8KCEr6b0PVnnUyaP9g"
}
```


Listing 5.1: Formato JSON recebido no Servidor de Rede

5.2 Testes à plataforma web

A plataforma web tem como objetivo fornecer a localização dos dispositivos da rede e analisar os valores obtido pelos mesmo recorrendo a uma interface simples. Estes dados, são posteriormente interpretados de modo a identificar um possível incêndio florestal e consequentemente realizar o procedimento de alerta já referido anteriormente. De modo a garantir este funcionamento mencionado, foram realizados testes onde é verificada a receção dos dados e a correta analise dos mesmos.

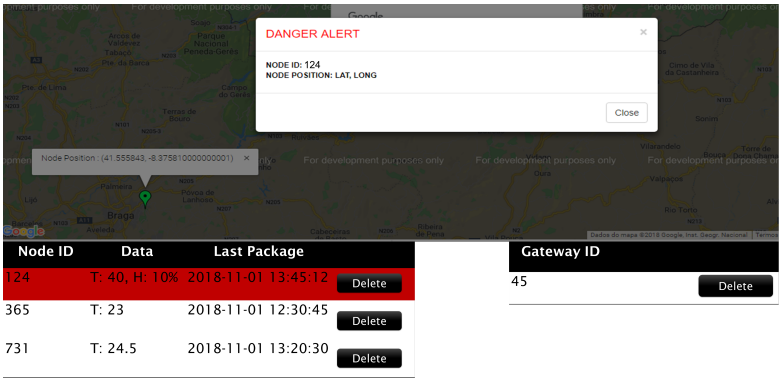


Figura 5.5: A imagem representa uma situação de alerta, em que um determinado nó (Node ID) deteta uma anomalia. É de imediato apresentada a localização do mesmo e assinalado na tabela o dispositivo em questão.

A figura 5.5, representa a plataforma web numa situação de alarme. Como é possível verificar, através da API Google Maps é assinalado o dispositivo que detetou o incêndio e posteriormente é indicada a localização do nó e o ID associado. Auxiliado desta representação gráfica mencionada, o sistema reproduz também um alarme sonoro capaz de alertar o utilizador. Este processo é possivel devido à utilização da função, JavaScript, *setInterval()* , que permite realizar a leitura dos dados presentes no servidor de rede e atualizar os mesmos como está representado no anexo B do presente documento. A partir da imagem em estudo, é de notar que os valores dos sensores presentes nos dispositivos estão a ser corretamente recebidos em tempo real garantido assim o funcionamento e propósito da plataforma web desenvolvida.

Capítulo 6

Conclusões e trabalho futuro

Após a análise detalhada de todo o projeto desenvolvido, revela-se fulcral realizar um levantamento das conclusões e possíveis melhorias relativas ao sistema apresentado no documento em questão. Assim o presente capítulo tem como objetivo abordar as diferentes conclusões obtidas após a finalização total do sistema e quais as perspetivas futuras a realizar de modo aprimorar o gateway.

6.1 Conclusões

Os objetivos estipulados foram cumpridos na totalidade, tendo sido obtido um protótipo capaz de realizar o reencaminhamento dos pacotes para um servidor de rede e a interpretação destes mesmos através de uma aplicação web. A possibilidade de realizar comunicação com o servidor e com o gestor da rede, recorrendo a SMS, através do módulo SIM800 foi também conseguida. O armazenamento de pacotes perdidos e o consequente reencaminhamento destes, também se revelaram eficazes. Deste modo, as funcionalidades básicas foram implementadas com sucesso e testadas conforme o capítulo dedicado aos mesmos.

Durante toda a fase de desenvolvimento as principais dificuldades foram encontradas na obtenção da PCB. Inicialmente, a secção responsável pelo funcionamento do Ethernet apresentava falhas tendo sido utilizado em alternativa o módulo ECN25j. Porém, numa segunda versão da PCB este obstáculo foi em grande parte ultrapassado conseguindo assim uma melhoria significativa no funcionamento desta secção. Relativamente à plataforma web, esta consegue realizar uma monitorização em tempo real e apresentar os dados através de uma interface simples. Os testes da plataforma, revelaram-se eficazes conseguindo alertar as unidades de combate de possíveis incêndios.

6.2 Trabalho Futuro

O gateway desenvolvido possui diversos periféricos entre eles HDMI. Este foi introduzido com o intuito de testar e verificar o correto funcionamento nos momentos de instalação e configuração

do OS. Porém, em futuros protótipos este periférico torna-se desnecessário conduzindo desta forma a uma diminuição de dimensões da *Costum Board*. Uma outra modificação futura que contribui para esta redução no tamanho da *board*, passa pelo posicionamento da *Radio Board*, horizontalmente, sobreposta à *Custom Board*. O gateway está em permanente execução e como tal necessita de uma alimentação constante. Para tal e tendo em conta ambientes *outdoors*, a incorporação de uma combinação de painel solar e bateria constitui também um trabalho futuro.

Relativamente ao software presente no gateway, este cumpre as necessidades básicas do mesmo realizando as funcionalidades impostas pelos requisitos referenciados anteriormente. Porém, a nível de segurança não é possível realizar *updates* ao próprio gateway ou aos nós da rede. Assim, uma futura modificação no software passa por possibilitar a realização de atualizações remotas nos dispositivos da rede assegurando desta forma um proteção dos mesmos.

No que concerne à aplicação desenvolvida para a realização da interpretação dos valores adquiridos e representação dos mesmos graficamente, esta requer também futuras modificações. A adição de uma camada dedicada à segurança, é uma incorporação imperativa no futuro de modo a evitar qualquer tipo de ataques à plataforma. O restante trabalho futuro da aplicação web, encontra-se relacionado com melhorias na interface de modo a tornar estas mais simples e possíveis modificações na base de dados após um estudo mais aprofundado da mesma (indo de encontro às necessidades das unidades de combate a incêndios).

Por último, será importante testar todo o sistema numa situação real, ou seja, inserir o gateway numa rede LoRa e sendo esta preferencialmente dedicada o controlo de incêndios. Apesar de até ao momento terem sido efetuados testes com dois nós equipados com sensores, revela-se importante testar o gateway na presença de um maior numero de nós e para assim se poder validar, de forma mais conclusiva, cada uma das áreas de desenvolvimento: hardware e software.

Anexos A

Phyton

Para os testes de alcance apresentados, foram utilizados algoritmos desenvolvidos em Python com o objetivo de realizar uma serialização da informação contida nos ficheiros JSON. O primeiro listing apresentado, tem como finalidade obter a percentagem de pacotes recebidos em cada uma das posições do nó, como foi apresentado no capítulo de testes na tabela xx.

```
1  import json
2  import base64
3
4  from collections import Counter
5  from pprint import pprint
6
7  y = 0
8  k = 0
9  sfi = 0
10 sf = 12
11 txs = []
12 sfs = [0,0,0,0,0,0]
13 txv =
       [[0,1,2,15,16,17,30,31,32,45,46,47,60,61,62,75,76,77],[3,4,5,18,19,20,33,34,35,48,
14
15 tx1 = [0,1,2,15,16,17,30,31,32,45,46,47,60,61,62,75,76,77]
16 tx2 = [3,4,5,18,19,20,33,34,35,48,49,50,63,64,65,78,79,80]
17 tx3 = [6,7,8,21,22,23,36,37,38,51,52,53,66,67,68,81,82,83]
18 tx4 = [9,10,11,24,25,26,39,40,41,54,55,56,69,70,71,84,85,86]
19 tx5 = [12,13,14,27,28,29,42,43,44,57,58,59,60,72,73,74,87,88,89]
20
21 with open('16Nov20181549.json') as f:
22   data = json.load(f)
23   c = Counter(k[0:] for d in data for k, v in d.items() if
         k.startswith('dev_id'))
24   l = Counter(d['metadata']['data_rate'] for d in data)
```

```
25
26
27 print(json.dumps(c, indent=2))
28 print(json.dumps(l, indent=2))
29
30 for d in data:
31   s = base64.b64decode(d['payload_raw']).encode("hex")
32   i = int(s, 16)
33   txs.append(i)
34
35
36
37 for tx in range(5):
38
39   for i in range(3):
40     if txv[tx][i] in txs:
41       k = k+1
42   print "SF", sf ,"Tx", tx+1 , k
43   sfs[sfi] = sfs[sfi]+k
44   sfi = sfi+1
45   k= 0
46   sf = sf - 1;
47
48   for i in range(3,6):
49     if txv[tx][i] in txs:
50       k=k+1
51   print "SF", sf ,"Tx",tx+1 , k
52   sfs[sfi] = sfs[sfi]+k
53   sfi = sfi+1
54   k = 0
55   sf = sf - 1;
56
57   for i in range(6,9):
58     if txv[tx][i] in txs:
59       k = k+1
60   print "SF", sf ,"Tx", tx+1 , k
61   sfs[sfi] = sfs[sfi]+k
62   sfi = sfi+1
63   k = 0
64   sf = sf - 1;
```

```
65
66    for i in range(9,12):
67      if txv[tx][i] in txs:
68        k = k+1
69    print "SF", sf ,"Tx", tx+1 , k
70    sfs[sfi] = sfs[sfi]+k
71    sfi = sfi+1
72    k= 0
73    sf = sf - 1;
74
75    for i in range(12,15):
76      if txv[tx][i] in txs:
77        k= k+1
78    print "SF", sf ,"Tx", tx+1 , k
79    sfs[sfi] = sfs[sfi]+k
80    sfi = sfi+1
81    k = 0
82    sf = sf - 1;
83
84    for i in range(15,18):
85      if txv[tx][i] in txs:
86        k = k+1
87    print "SF", sf ,"Tx", tx+1 , k
88    sfs[sfi] = sfs[sfi]+k
89    sfi = 0
90    k = 0
91    sf = 12
92
93  print sfs
94  print txs
95  for i in range(6):
96    print "SF", 12-i, (float(sfs[i])/15)*100, "%"
```

O segundo listing, tem como objetivo retirar os valores de RSSI e SNR dos ficheiros JSON e posteriormente inserir estes num ficheiro Excel.

```
1  import json
2  import xlwt
3
4  i = 0
```

```
5
6  #possivelmente da jeito SF, rssi, snr, freq
7
8  book = xlwt.Workbook(encoding="utf-8")
9
10 sheet1 = book.add_sheet("Sheet 1")
11
12 f = open('20181815.json', 'r')
13   distros_dict = json.load(f)
14
15 print "SNR:"
16
17 for distro in distros_dict:
18   sheet1.write(i, 0, distro['metadata']['gateways'][0]['rssi'])
19   i = i+1
20
21
22 i = 0
23
24 print "\nRSSI:"
25
26 for distro in distros_dict:
27   sheet1.write(i, 1, distro['metadata']['gateways'][0]['rssi'])
28   i = i+1
29
30 book.save("20181815rssi.xls")
```

Anexos B

JavaScript

```
1  $(document).ready(function(){
2    setInterval(get_server,xxxx); //Tempo de actulizacao da leitura dos
         dados
3  });
4                .
5                .
6                .
7   function get_server ()
8   {
9   console.log("Inside");
10  var markerId = 0;
11  var table = document.getElementById("nodes");
12  var tr = table.getElementsByTagName("tr");
13  var td, labled;
14  var marker;
15
16  $.ajax ({
17  type: 'GET',
18  url:'postfrom.php',
19  dataType: 'json',
20  success: function(response) {
21  //alert('Loaded!');
22  console.log(response);
23
24  for(var key in response)
25  {
26   markerId = response[key].ID;
27   var dados2 = response[key].dados;
28   console.log(tr);
29   for (i = 1; i < tr.length; i++)
```

```
30   {
31     td = tr[i].getElementsByTagName("td")[0].innerHTML;
32     console.log(td);
33     if(td == markerId)
34     {
35       marker = markers[markerId];
36       console.log (marker);
37       tr[i].getElementsByTagName("td")[1].innerHTML = response[key].dados
38     if(response[key].dados[1] > 10)
39     {
40       if(infoflag == 0)
41       {
42         var contentString = '<div id="google-popup" style = "color:
               black">' + '<p> Node Position : ' + marker.position +
               '</div>';
43         var infowindow = new google.maps.InfoWindow({
44         content: contentString
45       });
46         infowindow.open(map, marker);
47         infoflag = 1;
48       }
49       document.getElementById("myAudio").play();
50       document.getElementById("myAudio").muted = false;
51       $('#add_data_Modal3').modal('show');
52       document.getElementById("firea").innerHTML = "NODE ID:" +
             response[key].ID + "<br>NODE POSITION: LAT, LONG";
53     }
54   }
55   else
56   {soundflag = 0; infoflag = 0;}
57   break;
58     }
59    }
60   }
61  }
62  });
63  }
```

Listing B.1: Funçaõ utilizada na leitura dos dados.

Anexos C

Testes Iniciais

Os testes inicias tem como objetivo comprovar o funcionamento da *Radio Board* desenvolvida pela Microchip. Esta *board* encontra-se presente no *Evaluation Kit* 8000 tal como dois *end-devices* e uma *Core Board*, tendo como objetivo a criação de uma rede simples. Porém e como ja mencionado anteriormente, os componentes incluído neste kit não contemplam os objetivos desta dissertação conduzindo ao desenvolvimento de uma *Core Board* adequada. Contudo, numa fase inicial foram realizados teste recorrendo ao *Kit* 8000 de modo a comprovar o funcionamento dos componentes e do protocolo LoRaWAN.

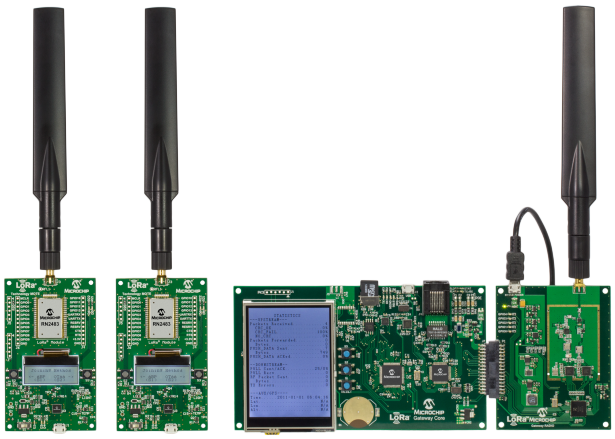


Figura C.1: Componentes presentes no Kit 800, dois end-devices e uma core board auxiliada da *radio board* já analisada anteriormente.

O *Evaluation Kit* fornece uma aplicação com a capacidade de interagir com os nós e gateway através de uma porta USB e Ethernet, possibilitando a configuração de chaves, ID dos dispositivos e parâmetros associado ao protocolo de forma simples. A aplicação, oferece também a possibilidade de realizar a leitura dos pacotes recebidos pelo gateway e consequentemente armazenar a informação da rede recorrendo a um servidor, fornecido também pela Microchip, presente num *container* Docker.

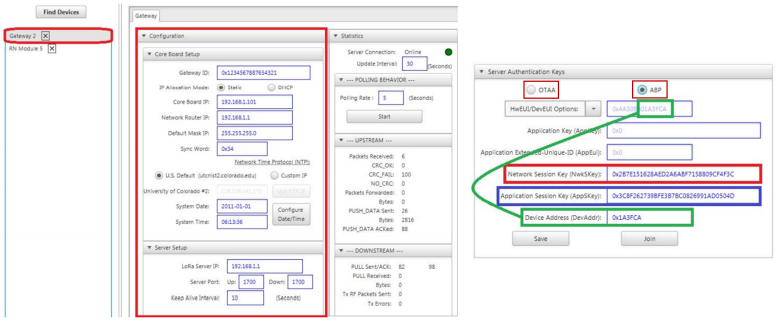


Figura C.2: Configurações de um gateway e end-device [47].

Para os testes em questão, foi inicialmente realizada a configuração do gateway, sendo posteriormente adicionado os dispositivos ao servidor através do *Device Address* (DevAddr) e das chaves NwSKey e AppSKey como esta representado na Figura C.2. Este ultimo procedimento difere consoante o método de ativação escolhido ABP ou OTAA, existindo na base de dados a distinção entre ambos. Após a conclusão da fase de inicialização, o sistema encontra-se preparado para receber pacotes provenientes dos *end-devices* sendo assim realizado o envio dos dados [47].

Os testes realizados permitiram aprofundar a compreensão relativa ao protocolo LoRaWAN e aos componentes que constituem a mesma. Apesar de este *Evaluation Kit* apresentar as condições para a construção de uma rede LoRa, verificou-se ao longo dos testes alguns erros de software que impedem a utilização do mesmo para o estudo da tecnologia LoRa. A constituição dos componentes como é o caso da *Core Board* presente do *Kit*, não contempla os objetivos da dissertação conduzindo à elaboração de uma nova tal como ja foi mencionado sendo o principal foco desta dissertação.

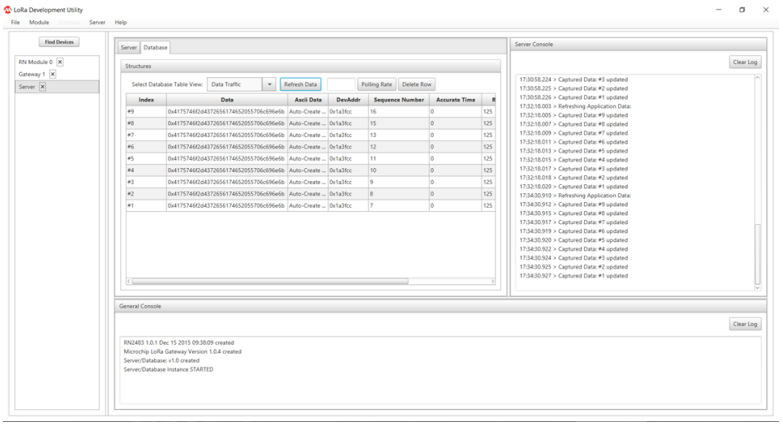


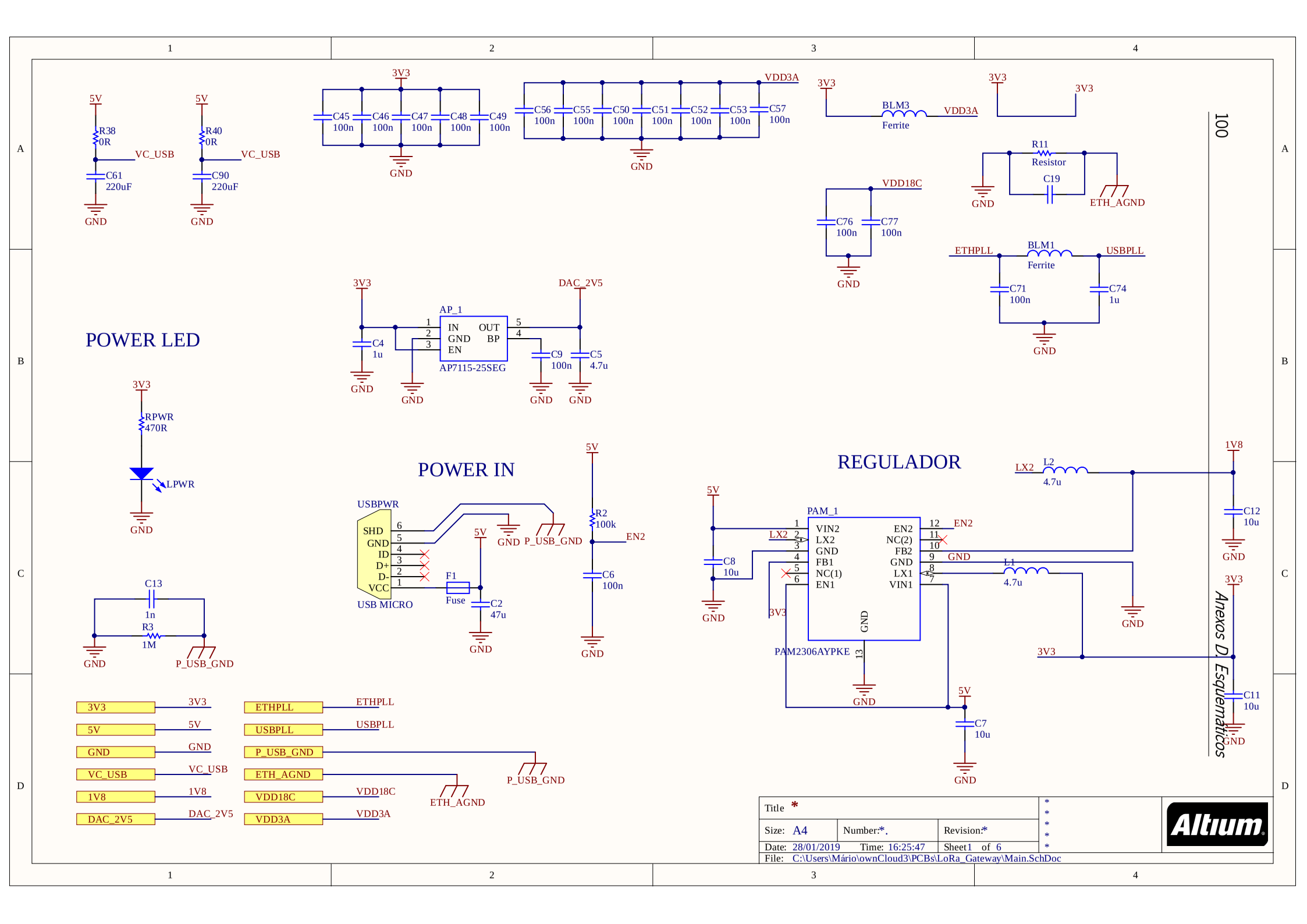
Figura C.3: Testes realizados com o Evaluation Kit 800.

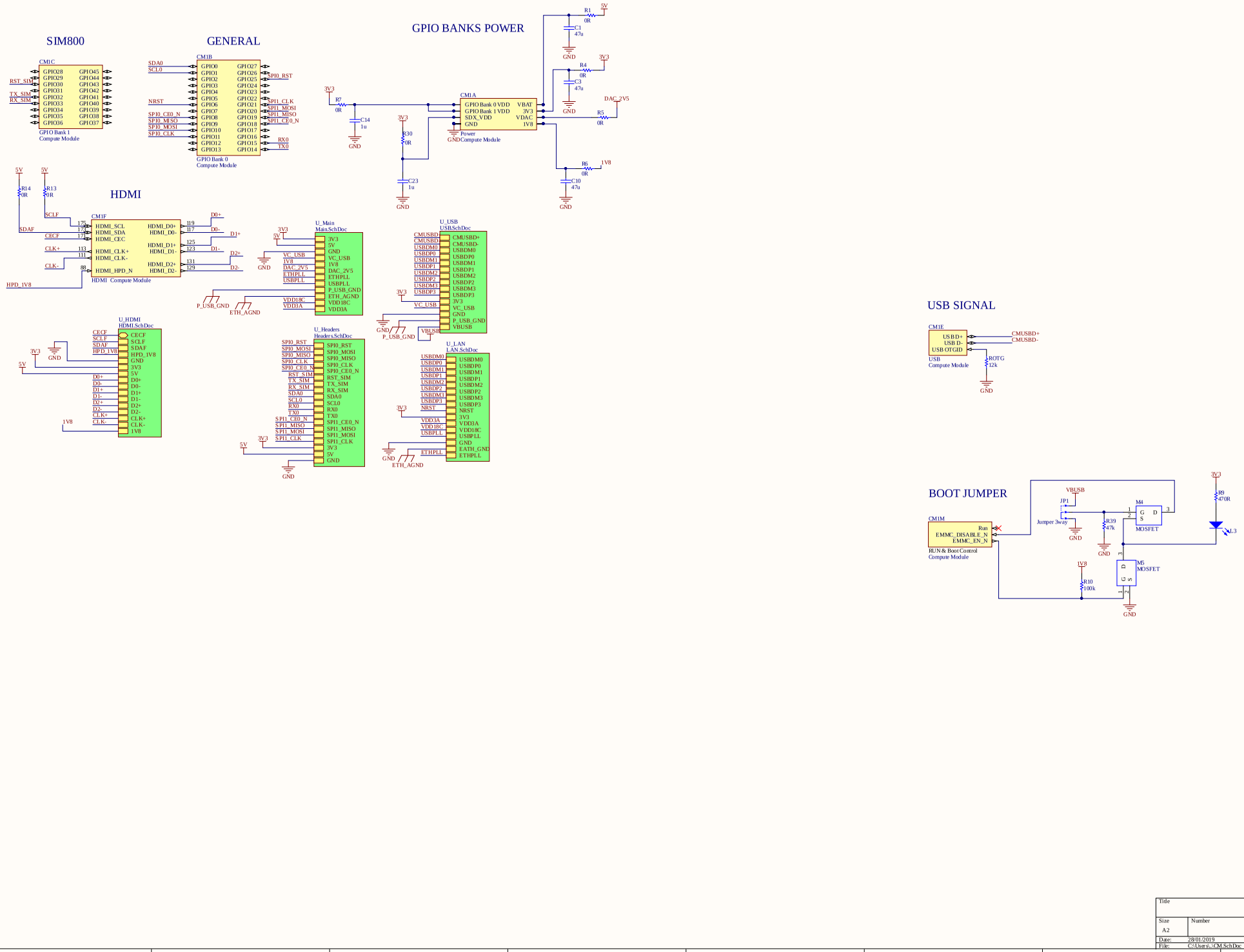
Os restantes testes iniciais realizados, tiveram como foco comprovar o funcionamento da *Radio Board* em três placas distintas: Raspberry Pi, STM32 e Pico-Pi iMX7. Para cada um dos

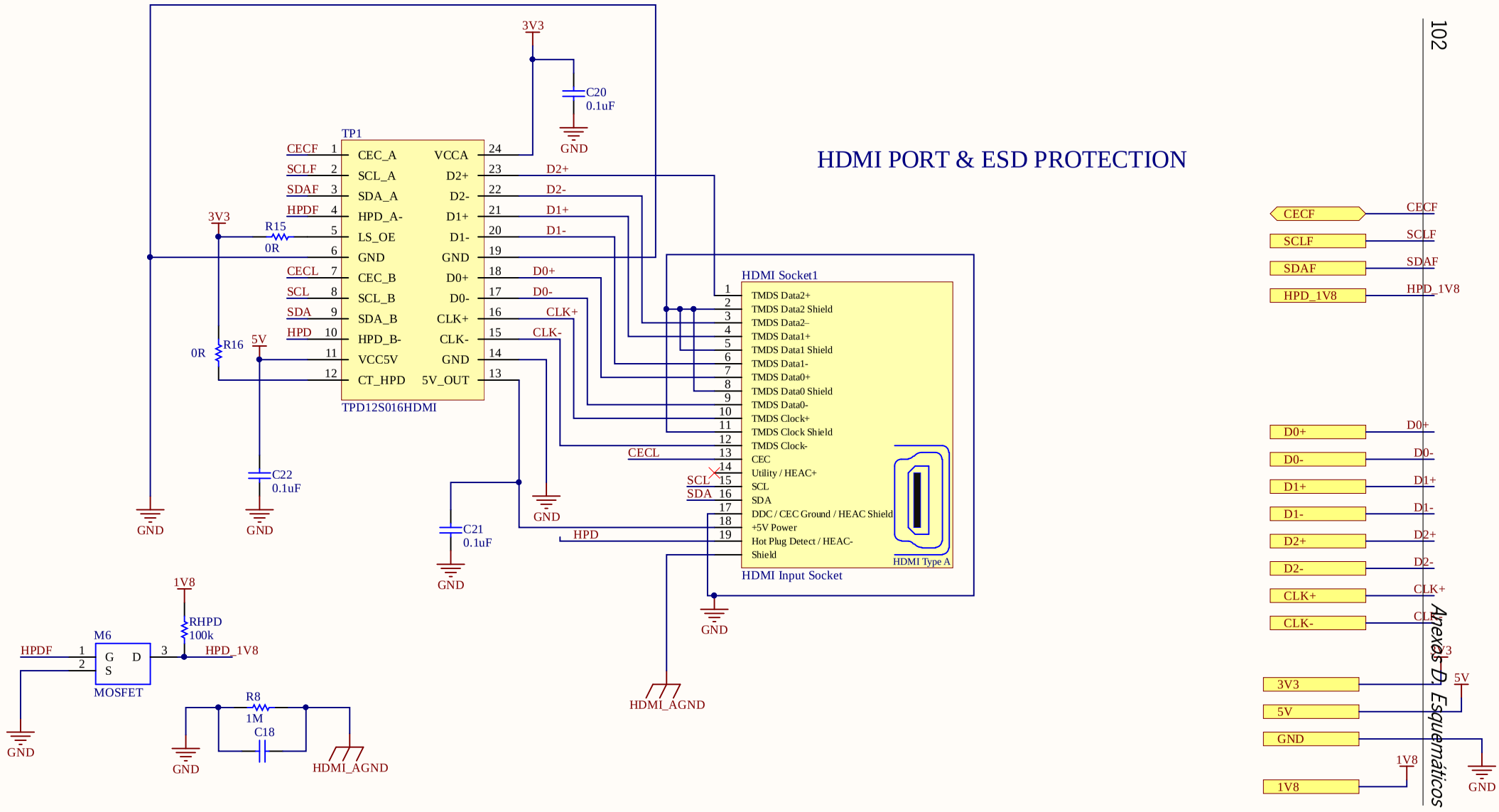
casos foram verificados problemas de ruídos na comunicação SPI entre as *boards* e a *Radio Board*, impedindo o funcionamento correto do gateway.

Anexos D

Esquemáticos

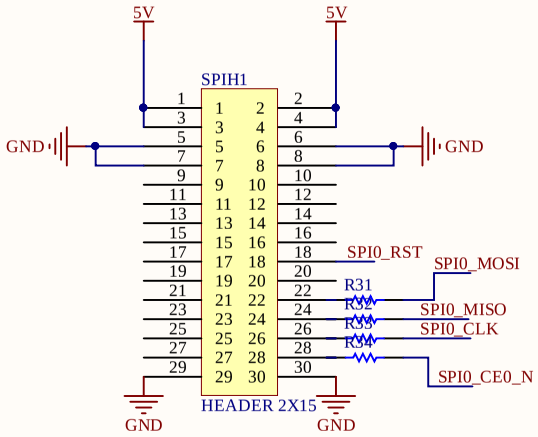

5V
R38
0R
VC_USB
C61
220uF
GND
5V
R40
0R
VC_USB
C90
220uF
GND
3V3
C45 100n
C46 100n
C47 100n
C48 100n
C49 100n
GND
VDD3A
C56 100n
C55 100n
C50 100n
C51 100n
C52 100n
C53 100n
C57 100n
GND
3V3
BLM3
Ferrite
VDD3A
3V3
3V3
R11
Resistor
C19
GND
ETH_AGND
VDD18C
C76 100n
C77 100n
GND
ETHPLL
BLM1
Ferrite
USBPLL
C71 100n
C74 1u
GND
100
POWER LED
3V3
RPWR
470R
LPWR
GND
C13
1n
R3
1M
GND
P_USB_GND
3V3
AP_1
IN
GND
EN
OUT
BP
AP7115-25SEG
C4
1u
C9
100n
C5
4.7u
DAC_2V5
GND
GND
GND
GND
POWER IN
USBPWR
SHD
GND
ID
D+
D-
VCC
USB MICRO
F1
Fuse
C2
47u
5V
GND
P_USB_GND
R2
100k
EN2
C6
100n
5V
GND
GND
REGULADOR
5V
C8
10u
GND
PAM_1
VIN2
LX2
GND
FB1
NC(1)
EN1
EN2
NC(2)
FB2
GND
LX1
VIN1
PAM2306AYPKE
LX2
3V3
EN2
GND
L1
4.7u
GND
LX2
L2
4.7u
1V8
C12
10u
GND
3V3
3V3
C11
10u
GND
5V
C7
10u
GND
3V3
5V
GND
VC_USB
1V8
DAC_2V5
ETHPLL
USBPLL
P_USB_GND
ETH_AGND
VDD18C
VDD3A
P_USB_GND
ETH_AGND
Anexos D. Esquemáticos
Title *
Size: A4
Number:*.
Revision:*
Date: 28/01/2019
Time: 16:25:47
Sheet1 of 6
File: C:\Users\Mário\ownCloud3\PCBs\LoRa_Gateway\Main.SchDoc
Altium

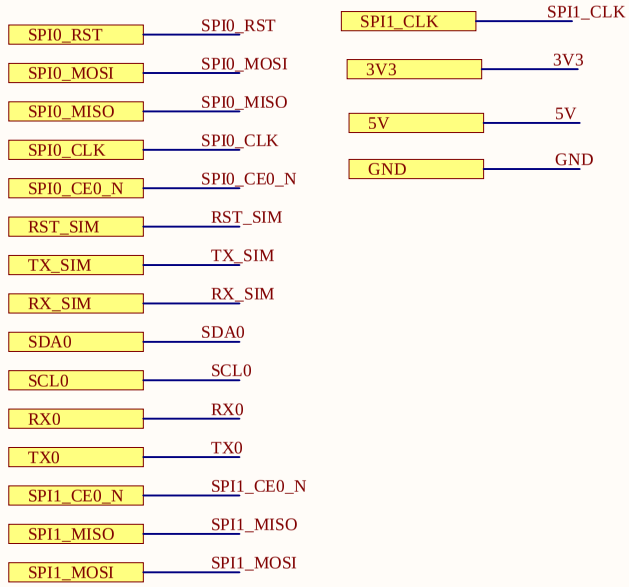

SIM800
GENERAL
GPIO BANKS POWER
HDMI
USB SIGNAL
BOOT JUMPER
CM1C
GPIO28 GPIO45
GPIO29 GPIO44
GPIO30 GPIO43
GPIO31 GPIO42
GPIO32 GPIO41
GPIO33 GPIO40
GPIO34 GPIO39
GPIO35 GPIO38
GPIO36 GPIO37
RST_SIM
TX_SIM
RX_SIM
GPIO Bank 1
Compute Module
CM1B
GPIO0 GPIO27
GPIO1 GPIO26
GPIO2 GPIO25
GPIO3 GPIO24
GPIO4 GPIO23
GPIO5 GPIO22
GPIO6 GPIO21
GPIO7 GPIO20
GPIO8 GPIO19
GPIO9 GPIO18
GPIO10 GPIO17
GPIO11 GPIO16
GPIO12 GPIO15
GPIO13 GPIO14
SDA0
SCL0
NRST
SPI0_CE0_N
SPI0_MISO
SPI0_MOSI
SPI0_CLK
SPI0_RST
SPI1_CLK
SPI1_MOSI
SPI1_MISO
SPI1_CE0_N
RX0
TX0
GPIO Bank 0
Compute Module
CM1A
GPIO Bank 0 VDD
GPIO Bank 1 VDD
SDX_VDD
GND
VBAT
3V3
VDAC
1V8
Power
GNDCompute Module
R1 0R
R4 0R
R5 0R
R6 0R
R7 0R
R30 0R
C1 47u
C3 47u
C10 47u
C14 1u
C23 1u
5V
3V3
1V8
DAC_2V5
GND
R14 0R
R13 0R
CM1F
HDMI_SCL
HDMI_SDA
HDMI_CEC
HDMI_CLK+
HDMI_CLK-
HDMI_HPD_N
HDMI_D0+
HDMI_D0-
HDMI_D1+
HDMI_D1-
HDMI_D2+
HDMI_D2-
HDMI Compute Module
SCLF
SDAF
CECF
HPD_1V8
D0+
D0-
D1+
D1-
D2+
D2-
CLK+
CLK-
U_Main
Main.SchDoc
VC_USB
ETHPLL
USBPLL
P_USB_GND
ETH_AGND
VDD18C
VDD33A
U_USB
USB.SchDoc
CMUSBD+
CMUSBD-
USBDM0
USBDP0
USBDM1
USBDP1
USBDM2
USBDP2
USBDM3
USBDP3
VBUSB
U_HDMI
HDMI.SchDoc
U_Headers
Headers.SchDoc
SPI0_MOSI
SPI0_MISO
SPI0_CLK
SPI0_CE0_N
RST_SIM
TX_SIM
RX_SIM
SDA0
SCL0
RX0
TX0
SPI1_CE0_N
SPI1_MISO
SPI1_MOSI
SPI1_CLK
U_LAN
LAN.SchDoc
NRST
VDD3A
VDD18C
USBPLL
EATH_GND
CM1E
USB D+
USB D-
USB OTGID
USB
Compute Module
ROTG 12k
CM1M
Run
EMMC_DISABLE_N
EMMC_EN_N
RUN & Boot Control
Compute Module
VBUSB
JP1
Jumper 3way
R39 47k
M4
MOSFET
M5
MOSFET
R9 470R
L3
R10 100k
Anexos D. Esquemáticos
101
Title
Size
A2
Number
Revision
Date: 28/01/2019
Sheet of
File: C:\Users\...\CM.SchDoc
Drawn By:



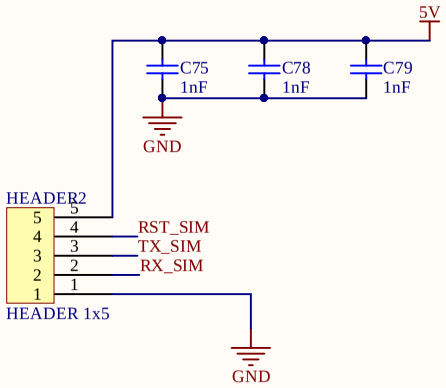
Title		
Size A4	Number	Revision
Date: 28/01/2019	Sheet of	
File: C:\Users\..\HDMI.SchDoc	Drawn By:	

SPI RADIO HEADER

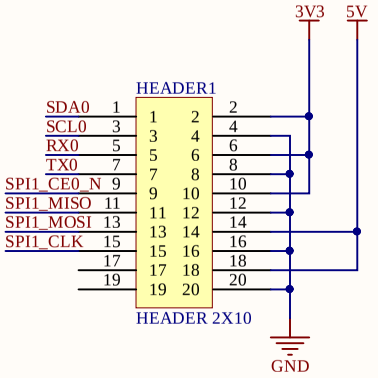




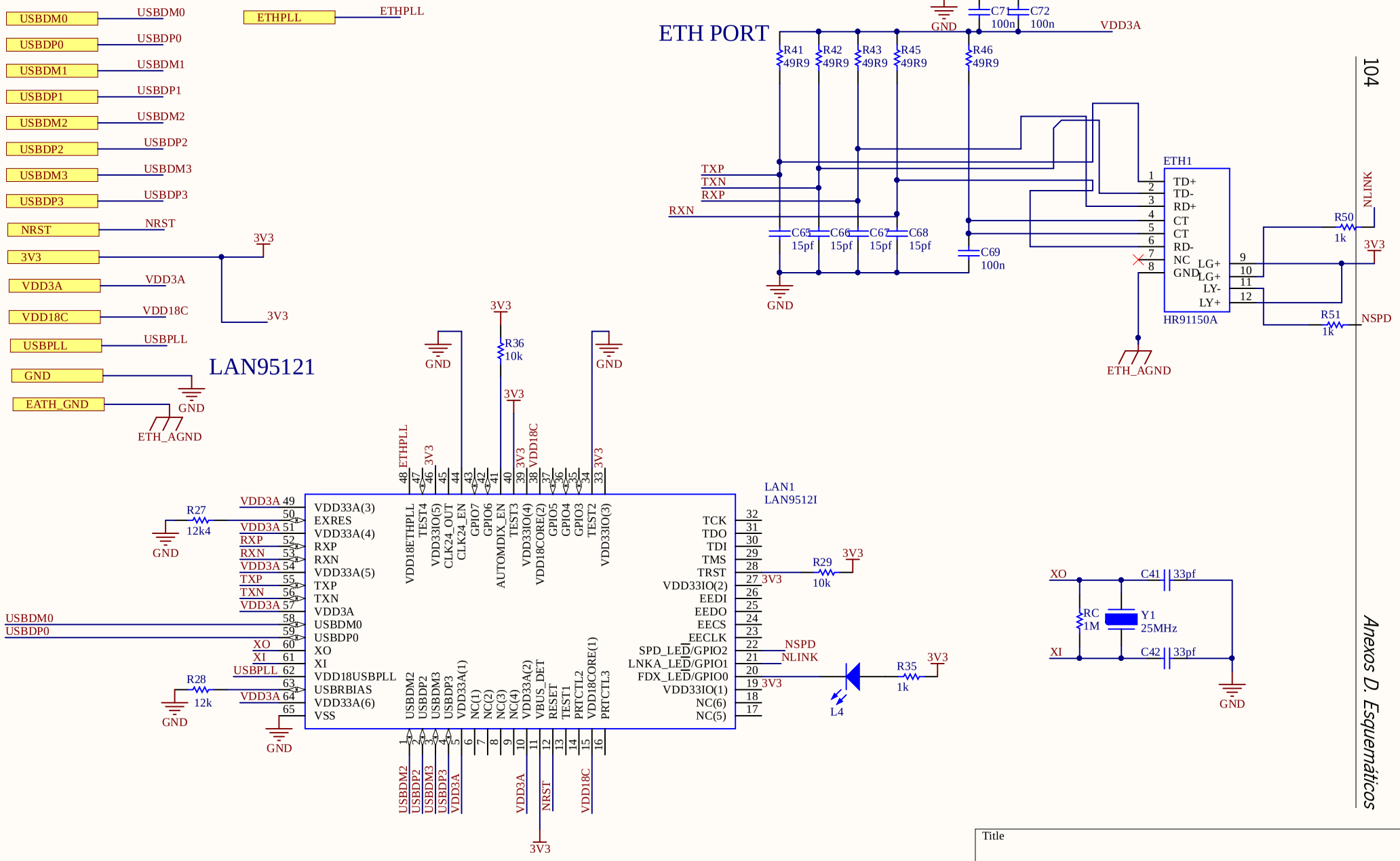
SIM800 HEADER



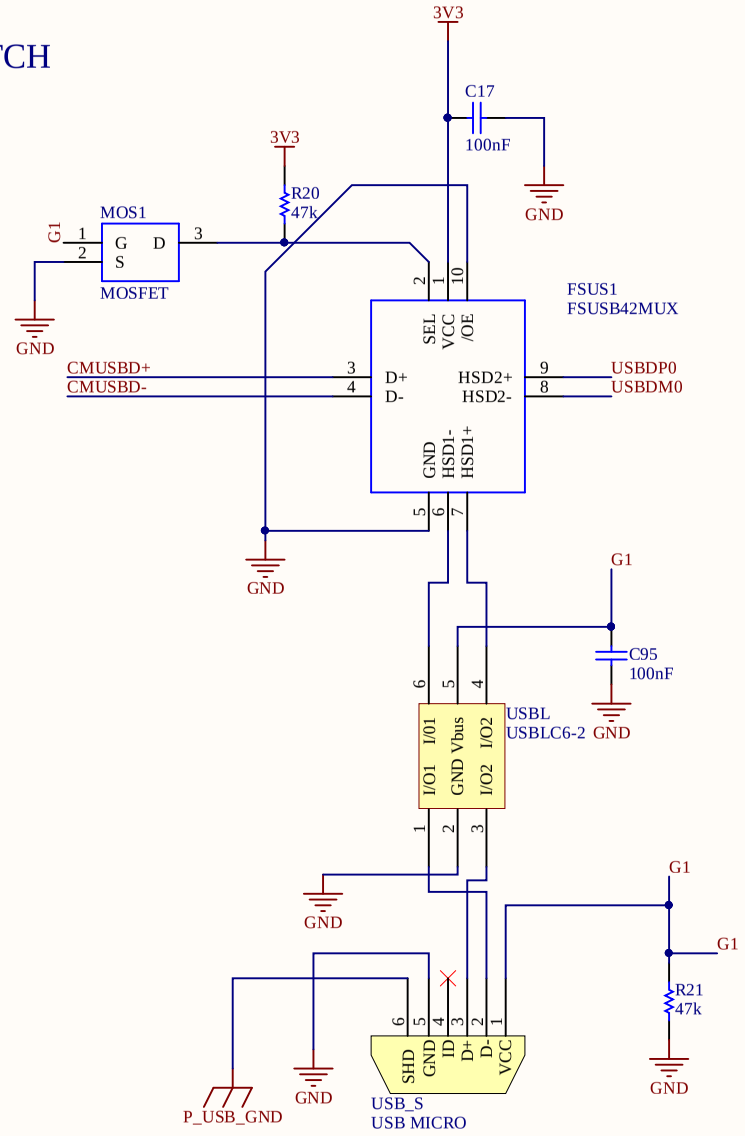
GENERAL HEADER



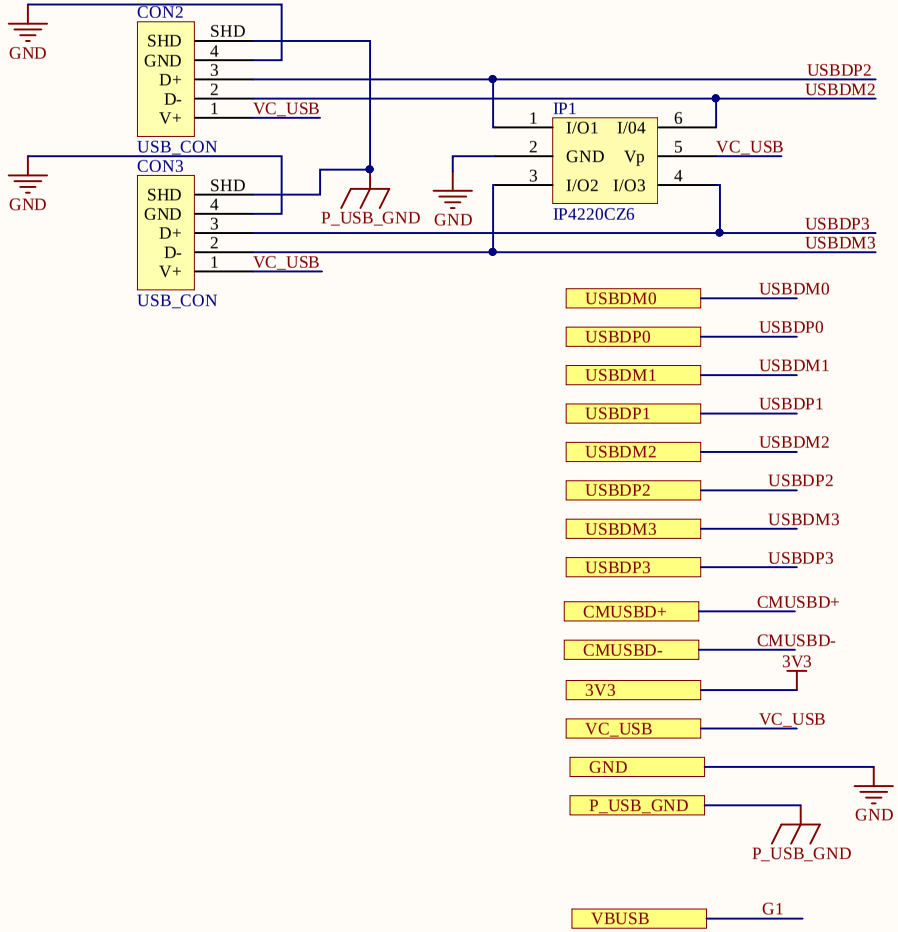
Title		
Size: A4	Number	Revision
Date: 28/01/2019		Sheet of
File: C:\Users\..\Headers.SchDoc		Drawn By:


ETH PORT
LAN95121
USBDM0
USBDP0
USBDM1
USBDP1
USBDM2
USBDP2
USBDM3
USBDP3
NRST
3V3
VDD3A
VDD18C
USBPLL
GND
EATH_GND
ETHPLL
C71 100n
C72 100n
VDD3A
R41 49R9
R42 49R9
R43 49R9
R45 49R9
R46 49R9
TXP
TXN
RXP
RXN
C65 15pf
C66 15pf
C67 15pf
C68 15pf
C69 100n
ETH1
TD+
TD-
RD+
CT
CT
RD-
NC
GND
LG+
LG+
LY-
LY+
HR91150A
ETH_AGND
R50 1k
R51 1k
NLINK
NSPD
R36 10k
R27 12k4
R28 12k
R29 10k
R35 1k
L4
LAN1
LAN9512I
VDD33A(3)
EXRES
VDD33A(4)
RXP
RXN
VDD33A(5)
TXP
TXN
VDD3A
USBDM0
USBDP0
XO
XI
VDD18USBPLL
USBRBIAS
VDD33A(6)
VSS
TCK
TDO
TDI
TMS
TRST
VDD33IO(2)
EEDI
EEDO
EECS
EECLK
SPD_LED/GPIO2
LNKA_LED/GPIO1
FDX_LED/GPIO0
VDD33IO(1)
NC(6)
NC(5)
RC 1M
Y1 25MHz
C41 33pf
C42 33pf
104
Anexos D. Esquemáticos
Title
Size A4
Number
Revision
Date: 28/01/2019
Sheet of
File: C:\Users\..\LAN.SchDoc
Drawn By:

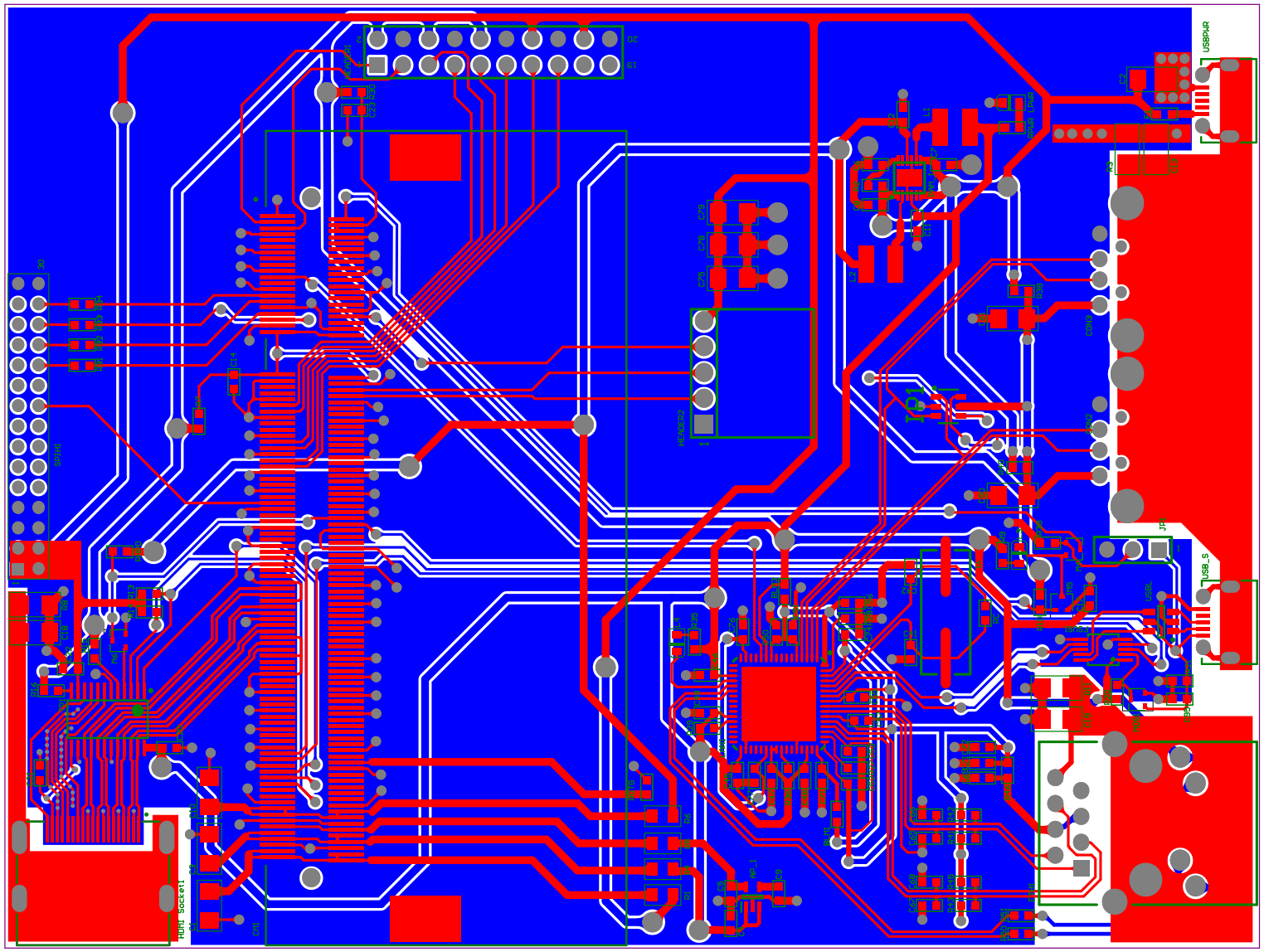
USB SWITCH



USB PORTS



Title		
Size: A4	Number	Revision
Date: 28/01/2019	Sheet of	
File: C:\Users\..\USB.SchDoc	Drawn By:	


USBPWR
USB_S
HDMI Socket1

Referencias

[1] "A brief history of the internet of things," Aug 2016. [Online]. Available: http://www.dataversity.net/brief-history-internet-things/

[2] D. Evans, "The Internet of Things - How the Next Evolution of the Internet is Changing Everything," *CISCO white paper*, no. April, pp. 1–11, 2011. [Online]. Available: http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:The+Internet+of+Things+-+How+the+Next+Evolution+of+the+Internet+is+Changing+Everything#0

[3] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 855–873, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7815384/

[4] "I-ON Communications Blog: LPWA (Low Power Wide Area), the core of IoT." [Online]. Available: https://ioncomm.blogspot.com/2016/10/lpwa-low-power-wide-area-core-of-iot.html [Accessed: 2017-12-13]

[5] "Incêndios florestais," Jun 2017. [Online]. Available: http://www.icnf.pt/portal/florestas/dfci/relat/rel-if

[6] A. Alkhatib, "A review on forest fire detection techniques," vol. 2014, 03 2013.

[7] LinkLab, "Low Power, Wide Area Networks networks (LPWANs)," 2017.

[8] W. Geographical, "Network Topologies Network Topologies Bus topology," *Access*, pp. 1–10.

[9] "Choosing lpwan technology for range," Mar 2016. [Online]. Available: https://iotbusinessnews.com/2016/03/25/30261-choosing-lpwan-technology-range/

[10] "Choosing lpwan technology for battery life," May 2016. [Online]. Available: https://iotbusinessnews.com/2016/04/22/54656-choosing-lpwan-technology-battery-life/

[11] "Choosing lpwan technology for quality of service," Mar 2016. [Online]. Available: https://iotbusinessnews.com/2016/03/11/50822-choosing-lpwan-technology-for-quality-of-service/

[12] "Lpwan technology for iot security," May 2016. [Online]. Available: https://iotbusinessnews.com/2016/05/06/37958-choosing-lpwan-technology-iot-security/

[13] "Choosing lpwan technology for network capacity," Feb 2016. [Online]. Available: https://iotbusinessnews.com/2016/02/26/16349-choosing-lpwan-technology-for-network-capacity/

[14] "A technical overview of LoRa ® and LoRaWAN ™," 2015. [Online]. Available: https://docs.wixstatic.com/ugd/eccc1a_ed71ea1cd969417493c74e4a13c55685.pdf

[15] "Sigfox Technical Overview," no. May, 2017.

[16] A. N. E. Guide, "AN EDUCATIONAL GUIDE An Educational Guide : How RPMA Works."

[17] "Rpma vs competition," 2017. [Online]. Available: https://www.ingenu.com/technology/rpma/competition/

[18] "Ingenu Node." [Online]. Available: https://www.ingenu.com/wp-content/uploads/2015/09/rACM-Ingenu-Transparent-Medium-1024x749.png

[19] "LoRa Device." [Online]. Available: https://www.cooking-hacks.com/media/cooking/images/documentation/tutorial_kit_lorawan/lorawan_representative_big.jpg

[20] "Sigfox Node." [Online]. Available: https://storage.sbg1.cloud.ovh.net/v1/AUTH_669d7dfced0b44518cb186841d7cbd75/prod_medias/j1zgxc8m.jpeg

[21] "Css image," 2017. [Online]. Available: http://www.newwaveinstruments.com/resources/reprints/primers/tutorials/spread_spectrum_communications/project.htm [Accessed: 2018-01-18]

[22] W. Sussex, "A Comparison of UNB and Spread Spectrum Wireless Technologies as used in LPWA M2M Applications A white paper by Real Wireless," 2015.

[23] "Signal CSS." [Online]. Available: https://www.sure-fi.com/images/chirp.png [Accessed: 2019-06-19]

[24] "LoRa MAC." [Online]. Available: https://www.rs-online.com/designspark/rel-assets/ds-assets/uploads/images/570641302c044bf0ac2734360ab5c436LoRaWAN_Classes.jpg [Accessed: 2019-06-19]

[25] LoRaAlliance and LoRa Alliance, "LoRaWAN 1.1 Specification," *LoRa Alliance*, no. 1.1, p. 101, 2017. [Online]. Available: https://lora-alliance.org/resource-hub/lorawantm-specification-v11

[26] M. Bor and U. Roedig, "LoRa transmission parameter selection," *Proceedings - 2017 13th International Conference on Distributed Computing in Sensor Systems, DCOSS 2017*, vol. 2018-January, pp. 27–34, 2018.

[27] "WIRELESS & SENSING," 2013. [Online]. Available: http://www.semtech.com/images/datasheet/LoraDesignGuide_STD.pdf

[28] "LoRa SF and DR." [Online]. Available: https://www.researchgate.net/profile/Mj_thinus_Booysen/publication/324043563/figure/fig7/AS:668539769851904@1536403710023/LoRa-Spreading-Factor-SF-Bitrates-and-Time-on-Air-LoRa-is-chosen-as-the-wireless.ppm [Accessed: 2019-06-19]

[29] "RSSI and SNR." [Online]. Available: https://i2.wp.com/electronicsforengineer.com/wp-content/uploads/2019/02/SNR-RSSI2.png?ssl=1 [Accessed: 2019-06-19]

[30] LoRa Alliance, "LoRaWAN Security Full End–to–End Encryption For IoT Application Providers," *LoRa Alliance - LoRaWAN Specification*, no. February, p. 4, 2016. [Online]. Available: https://www.lora-alliance.org/What-Is-LoRa/LoRaWAN-White-Papers%0Ahttps://www.lora-alliance.org/portals/0/documents/whitepapers/LoRaWAN_Security-Whitepaper_V6_Digital.pdf

[31] I-SCOOP, "LoRa and LoRaWAN: the technologies, ecosystems, use cases and market." [Online]. Available: https://www.i-scoop.eu/internet-of-things-guide/iot-network-lora-lorawan/ [Accessed: 2018-12-04]

[32] Semtech, "Smart Metering | LoRa Applications | Semtech LoRa Technology | Semtech." [Online]. Available: https://www.semtech.com/lora/lora-applications/smart-metering [Accessed: 2018-12-04]

[33] Semtech, "Smart Supply Chain & Logistics | LoRa Applications | Semtech LoRa Technology | Semtech." [Online]. Available: https://www.semtech.com/lora/lora-applications/smart-supply-chain-logistics [Accessed: 2018-12-04]

[34] Semtech, "Smart Agriculture | LoRa Applications | Semtech LoRa Technology | Semtech." [Online]. Available: https://www.semtech.com/lora/lora-applications/smart-agriculture [Accessed: 2018-12-04]

[35] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melià-Seguí, and T. Watteyne, "Understanding the Limits of LoRaWAN." [Online]. Available: https://arxiv.org/pdf/1607.08011.pdf

[36] M. Cattani, C. A. Boano, and K. Römer, "An Experimental Evaluation of the Reliability of LoRa Long-Range Low-Power Wireless Communication," *Journal of Sensor and Actuator Networks*, vol. 6, no. 2, p. 7, 2017. [Online]. Available: http://www.mdpi.com/2224-2708/6/2/7

[37] The Things Network, “Limitations | The Things Network.” [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/limitations.html [Accessed: 2018-12-04]

[38] R. Miller, “LoRa Security.”

[39] L. Ip, “User manual,” pp. 1–39.

[40] C. Lora and S. Kit, “MultiConnect ® Conduit ™ Lora ™ Starter Kit. ®.”

[41] H. K. Features, “Wirnet Station 868.”

[42] Lorrier, “Lorrier LR2.” [Online]. Available: https://lorrier.com/#introducing-lr2 [Accessed: 2018-12-04]

[43] R. Pi, “Raspberry Pi Compute Module (CM1) Raspberry Pi Compute Module 3 (CM3) Raspberry Pi Compute Module 3 Lite (CM3L),” vol. 3, no. June, 2018.

[44] U. S. B. Hub and E. Controller, “LAN9514/LAN9514i USB 2.0 Hub and 10/100 Ethernet Controller Data Sheet,” pp. 1–54, 2016.

[45] Microchip Technology Inc., “ENC28J60 Data Sheet Stand-Alone Ethernet Controller with SPI ™ Interface,” p. 102, 2004.

[46] SIMCom, “SIM800L_Hardware_Design_V1.00,” pp. 1–70, 2013. [Online]. Available: http://wiki.seeedstudio.com/images/4/46/SIM800L_Hardware_Design_V1.00.pdf

[47] *LoRa ® Technology Evaluation Suite User’s Guide*, 2016. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/40001847A.pdf

[48] M. Pérez, “SX1255 SX1257 RF Front-End Transceiver Low Power Digital I and Q RF Multi-PHY Mode Transceiver,” *WD info*, no. February, p. 2004, 2003.

[49] Semtech Corporation, “WIRELESS & SENSING PRODUCTS Datasheet SX1301,” no. May, pp. 1–40, 2017.

[50] The Things Network, “Network | The Things Network.” [Online]. Available: https://www.thethingsnetwork.org/docs/network/ [Accessed: 2018-12-10]

[51] The Things Network, “Applications | The Things Network.” [Online]. Available: https://www.thethingsnetwork.org/docs/applications/

[52] The Things Network, “Overview | The Things Network.” [Online]. Available: https://www.thethingsnetwork.org/docs/applications/integrations.html

[53] D. Incorporated, “Dual High-Efficiency Pwm Step-Down Dc-Dc Converter,” no. November, pp. 1–15, 2012. [Online]. Available: www.diodes.com

[54] N. O. T. Recommended and F. O. R. New, “AP7115 AP7115 Pin Descriptions,” no. September, pp. 1–12, 2017.

[55] Raspberry Pi, “Device Trees, overlays, and parameters - Raspberry Pi Documentation.” [Online]. Available: https://www.raspberrypi.org/documentation/configuration/device-tree.md [Accessed: 2018-12-05]

[56] Semtech, “Packet Fowarder by Semtech,” pp. 1–3, 2017. [Online]. Available: https://github.com/Lora-net/packet_forwarder